mikroklan

ZESZYT 1

Scanned
by
Hawk

ENTER

ISSN 0860-1941 WYDAWNICTWO NOT SIGMA CENA 200 zł

Historia maszyn cyfrowych liczy niewiele ponad 30 lat. Należy się więc dziwić, że w skali tak krótkiego czasu, można już mówić o epokowym zacofaniu polskiej informatyki. Tłumaczyć można to tym, że jest to dziedzina rozwijająca się w tempie, jakiego nie zna dotychczasowa historia ludzkości. Rewelacja sprzed kilku zaledwie lat, dziś jest już muzealnym eksponatem. Narzucone przez przodujące kraje, mordercze tempo postępu pozostawia innych praktycznie bez szans. Bez szans w produkcji układów i sprzętu mikrokomputerowego, ale nie w jego wykorzystaniu. Rozwój mikroelektroniki sprawił bowiem, że sprzęt informatyczny stał się względnie tani i powszechnie dostępny – przynajmniej dla obywateli krajów rozwiniętych. Po zniesieniu barier celnych trafia on także do Polski. Naszej szansy należy więc upatrywać w umiejętnym zastosowaniu tego sprzętu, stworzeniu odpowiedniego oprogramowania i ewentualnej rozbudowie, tak by mógł on sprostać zadaniom, jakie stawia współczesne społeczeństwo.

Pierwszy krok został zrobiony: tysiące ZX Spectrum i Commodore trafiło do prywatnych domów, instytutów i zakładów pracy. Niestety, brak informatycznych tradycji sprawia, że niewielu użytkowników potrafi zastosować mikrokomputer jako narzędzie pracy. Na razie dominują gry i jeśli pozostaną one jedynym zadaniem dla mikrokomputerów, to wypadnie stwierdzić, że zafundowaliśmy sobie dość drogie zabawki.

Chociaż mikrokomputerom nadano nazwę „osobiste", ich użytkownicy niewiele osiągną działając w izolacji. Powstają więc różne grupy zainteresowanych, a nawet kluby – często nieformalne, o ograniczonym, niestety, zasięgu oddziaływania. Nie ma u nas, jak dotąd, żadnego forum dla szerszej wymiany doświadczeń, pomysłów i poglądów. Stąd tak niewiele zastosowań mikrokomputerów wykracza poza przywiezione z Zachodu gry.

Mikrokomputerowa moda pociągnęła za sobą jedynie zalew niepełnych, zniekształconych, czasem wręcz fałszywych informacji – na dodatek porozrzucanych po różnych pismach. Pismach, które nigdy przedtem nie miały nic wspólnego nie tylko z informatyką, ale nawet z techniką. Nie sposób w tej sytuacji przecenić wagi trzeźwego spojrzenia mikrokomputerowego praktyka.

Wprowadzona w latach 1984-1985 do miesięcznika INFORMATYKA wkładka MIKROKLAN była próbą uzdrowienia istniejącej sytuacji na rynku wydawniczym. Uznanie, jakie zdobył MIKROKLAN skłoniło redakcję do podjęcia starań o rozszerzenie zakresu i zasięgu tej publikacji. Inicjatywa ta znalazła poparcie Wydawnictwa NOT-Sigma, które uruchamia od 1986 r. serię wydawniczą „MIKROKLAN – zastosowanie i eksploatacja mikrokomputerów". Przekazując Czytelnikom pierwszy zeszyt tej serii, liczymy, że w bieżącym roku wydamy łącznie 10-12 zeszytów.

Oprócz materiałów autorskich, krajowych, znaczną część zawartości wypełnią publikacje z prasy zagranicznej, głównie z zachodnioniemieckiego miesięcznika MICRO. Możliwość taka powstała dzięki nawiązaniu współpracy z redakcją austriackiej mutacji tego miesięcznika. Zapewni to szybki dopływ informacji, co w przypadku techniki mikrokomputerowej jest sprawą o kapitalnym znaczeniu. Istotnym czynnikiem przyspieszającym ten dopływ jest możliwość terminowego drukowania za granicą, w dużym nakładzie, na miarę istniejącego zapotrzebowania społecznego.

Sądzimy, że staranny i fachowy dobór materiałów oraz szybkość procesu wydawniczego MIKROKLANU, umożliwią nadążanie, przynajmniej w obszarze informowania, za obecnym światowym tempem rozwoju techniki mikrokomputerowej.

Otwieramy nasze łamy nie dla bałwochwalczych zachwytów nad mikrokomputerem, lecz dla pokazania, w jakich sytuacjach może być on naprawdę przydatny i jak spowodować by sprostał naszym potrzebom i wymaganiom. Liczymy, że nasze wydawnictwo ułatwi obieg informacji między rozproszonymi dotąd użytkownikami, tworząc pewną wspólnotę ludzi zainteresowanych racjonalnym wykorzystaniem mikrokomputerów. Grupę bez prezesów, legitymacji i składek – połączoną tylko wspólną ideą.

Wierzymy, że MIKROKLAN będzie się szybko poszerzał. By dać szansę nowicjuszom, przewidujemy publikację wielu materiałów na poziomie elementarnym. Będzie więc mowa o różnych zastosowaniach mikrokomputera (np. w biurze, pracowni projektowej), programowaniu (na początek kurs języka BASIC) i najprostszych konstrukcjach sprzętowych. Będziemy starać się aby każdy, niezależnie od stopnia zaawansowania, znalazł tu dla siebie zawsze coś interesującego.

Redakcja

mikroklan ZESZYT 1

Wprowadzenie
w zagadnienia
techniki
mikrokomputerowej

Praca zbiorowa

Opracował zespół redakcji
miesięcznika INFORMATYKA

Teresa Jabłońska, Władysław Klepacz,
Ireneusz Myzik, Jerzy Orkiszewski,
Andrzej J. Piotrowski, Adam Pluta,
Zbigniew Pojmański, Jakub Tatarkiewicz,
Danuta Sot (oprac. techniczne),
Mateusz Stryjecki (oprac. graficzne)

WYDAWNICTWO CZASOPISM
I KSIĄŻEK TECHNICZNYCH
SIGMA
PRZEDSIĘBIORSTWO NACZELNEJ
ORGANIZACJI TECHNICZNEJ

00-950 Warszawa, skrytka 1004
ul. Biała 4

ISSN 0860-1941
Wydawnictwo Czasopism
i Książek Technicznych
NOT – Sigma
Warszawa 1986

Redakcja: 00-041 Warszawa,
ul. Jasna 14/16 p.244 tel. 27-71-40
Skład: techniką fotoskładu Eurocat 150
Wydawnictwo NOT-Sigma
Druk: Bohmann Druck und Verlag
GmbH & Co. KG., Wiedeń, Austria
Nakład: 100 000 egz.
Cena 200 zł

SPIS TREŚCI

SPRZĘT

- Programator pamięci EPROM
  Stanisław Gardynik 18
- Katarynka
  Jerzy Orkiszewski 23
- Jak dołączyć 2048 portów do ZX Spectrum
  Krzysztof Kontek 24
- Generator znaków
  Jerzy Orkiszewski 31

OPROGRAMOWANIE

- Bez stresów
  micro 10
- Jeden dla wszystkich
  micro 11
- Czym jest Unix (1). System plików
  Janusz Zalewski 21

ZASTOSOWANIE

- ZX Spectrum
  Andrzej J. Piotrowski 30

INFORMATYCZNE ABC

- BASIC dla początkujących (1)
  micro 2
- Pierwsze kroki...
  micro 4
- Jak dbać o dyskietki?
  micro 8

EDUKACJA

- Zaślepienie czy zacofanie
  oprac. *Katarzyna Isaak* 26

OPIS

- Trzy i półcalowe niemowlę
  micro 6
- Amstrad story
  Marek Górecki 14

KORESPONDENCJA

- Jak pisać do MIKROKLANU 32

- Bit za trzy grosze
  Jakub Tatarkiewicz 28

NOWOŚCI 3, 7, 13, 27

CENY 32

BASIC DLA POCZĄTKUJĄCYCH

Posługiwanie się wyłącznie gotowymi (zakupionymi) programami prowadzi użytkownika mikrokomputera do frustracji. Prawie każdego świerzbią palce, aby samemu napisać choć jeden program. Nasz kurs pomoże Ci zrealizować ten zamiar.

Kurs został opracowany dla użytkowników mikrokomputera Commodore 64, chociaż może być łatwo przeniesiony na niemal wszystkie inne mikrokomputery. W osobnym komentarzu znajdą się uwagi o istotnych różnicach dla ZX Spectrum.

Będziemy posuwać się naprzód stopniowo: w każdym odcinku kilka kolejnych instrukcji. Będą też ciekawostki, rady i użyteczne sztuczki.

Zanim włączysz mikrokomputer powinieneś się dowiedzieć, że może on pracować w trzech różnych trybach: bezpośrednim, edycji i wykonywania programu.

Nie irytuj się z powodu nowych pojęć; wytłumaczymy wszystko, posługując się łatwymi do przyswojenia przykładami. Na początek zajmijmy się trybem bezpośrednim, nie jest on niczym innym niż wykorzystaniem mikrokomputera jako kalkulatora kieszonkowego.

MIKROKOMPUTER JAKO KALKULATOR

Po włączeniu zasilania na ekranie pojawia się tak zwane *zgłoszenie systemowe mikrokomputera*. Całego zgłoszenia nie będziemy tu dokładnie omawiać. W widocznym na ekranie zgłoszeniu systemowym najistotniejsze jest słowo *READY*, które oznacza, że mikrokomputer jest gotowy do przyjmowania dalszych poleceń.

Zadaj teraz mikrokomputerowi najprostsze pytanie: ile jest 2 + 2. W tym celu wciśnij kolejno klawisze:

?2+2

a na koniec klawisz *RETURN*. W następnym wierszu w drugiej kolumnie pojawi się *4*. Pierwsza kolumna zawiera znak (w tym przypadku dodatni).

Uwaga: Liczby zawsze przedstawiane są ze znakiem, przy czym znak + pojawia się jako spacja (znak pusty, odstęp).

Uwaga ta okaże się bardzo ważna przy późniejszym formatowaniu danych wyjściowych. Zwróć uwagę, że pod czwórką pojawił się znowu komunikat *READY*, a pod nim miga kursor (znacznik pisania). Mikrokomputer oczekuje na dalsze polecenia. Wprowadź:

?2★2

i znów *RETURN*. Na ekranie pojawia się wynik.

Wyrażenia matematyczne możesz wprowadzać do mikrokomputera w ten sam sposób, jak byś je zapisywał na papierze. Dozwolone jest także używanie nawiasów. Spróbujemy jeszcze wykonać dwa pozostałe podstawowe działania arytmetyczne, wprowadzając:

?2–2 (klawisz RETURN)
?2/2 (klawisz RETURN)

W obu przypadkach mikrokomputer (jeśli tylko nie jest zepsuty...) podaje prawidłowy wynik, a po nim komunikat *READY*.

Przećwiczyłeś już z mikrokomputerem cztery podstawowe działania arytmetyczne i wiesz, że na klawiaturze odpowiadają im następujące symbole: + dodawanie, – odejmowanie, ★ mnożenie, / dzielenie.

Prawdopodobnie spotkałeś się w wyrażeniach matematycznych z oznaczeniem *AB* przy mnożeniu *A* przez *B*. W mikrokomputerze jest to niedozwolone. Przy mnożeniu zawsze musi być użyty znak ★.

NAWIASY

Jak już wspomnieliśmy mikrokomputer interpretuje również nawiasy; wprowadź

?2★(3+4)

i wciśnij *RETURN*. Spójrz czy wynik się zgadza. Jak dotąd mikrokomputer liczy prawidłowo.

KLAWISZE STERUJĄCE POŁOŻENIEM KURSORA

Teraz, gdy już sobie nieco postukałeś w klawisze, idziemy dalej. Z prawej strony u dołu klawiatury znajduje się klawisz z napisem *CRSR*. Za jego pomocą można przesuwać kursor (znacznik pisania): w prawo – gdy wciśniesz tylko klawisz *CRSR* lub w lewo – gdy wciśniesz równocześnie klawisze *CRSR* i *SHIFT*. Jeśli naciśniesz klawisz tylko przez chwilę – kursor na ekranie przesunie się o jedną kolumnę, jeśli dłużej – kursor zacznie wędrować dalej w wybranym kierunku, tak jakby następowały kolejne wciśnięcia. Zatem klawisz *CRSR* jest klawiszem powtarzającym.

Gdy napisałeś jakiś tekst błędnie, naprowadź kursor na źle wprowadzone znaki i na ich miejsce wstaw prawidłowe.

Co jednak zrobić, gdy w którymś z rzędów wpisałeś zbyt wiele lub zbyt mało znaków? Mikrokomputer pomoże Ci i w tym przypadku. Z prawej strony u góry klawiatury znajduje się klawisz *INST/DEL*. Za jego pomocą znaki mogą być usunięte (bez użycia klawisza SHIFT) lub wstawione (z użyciem klawisza SHIFT). Oznaczenie klawisza pochodzi z języka angielskiego:
INSERT – wstawiać i *DELETE* – usuwać.

Jeśli chcesz usunąć jakiś znak, ustaw kursor za znakiem przeznaczonym do usunięcia i wciśnij klawisz *INST/DEL*. Gdy chcesz wstawić znak, ustaw kursor na pozycję znaku przeznaczonego do wstawienia, wciśnij klawisz *SHIFT* i przy wciąż wciśniętym klawiszu *SHIFT*, wciśnij jeden raz klawisz *INST/DEL*.

Można również przesuwać kursor w górę i na dół; służy do tego drugi klawisz od prawej strony w dolnym rzędzie, który też jest oznaczony symbolem *CRSR* z odpowiednio skierowanymi strzałkami. Także i tutaj, w celu wyboru kierunku, należy użyć klawisz *SHIFT* lub go pominąć. Gdy już poznasz nieco klawiaturę i możliwości wprowadzania znaków do mikrokomputera, spróbuj wykonać kilka obliczeń – z użyciem nawiasów i bez.

Pokażemy teraz, że mikrokomputer ma nieco większe możliwości niż tylko wykonywanie czterech podstawowych działań. Wprowadź:

?2↑2

Klawisz ze strzałką skierowaną do góry znajduje się z prawej strony klawiatury. Wciśnij *RETURN*. Z pewnością nasunęła Ci się już następująca uwaga: wciśnięcie klawisza *RETURN* powoduje zakończenie wprowadzania znaków do mikrokomputera.

Przekonamy się jeszcze, że wciśnięcie klawisza *RETURN* nie jest jedyną możliwością zakończenia operacji wprowadzania. Wprowadź teraz:

?2↑6 [RETURN]

Jak widzisz, strzałka skierowana ku górze jest w mikrokomputerze operatorem potęgowania. Za jej pomocą wprowadza się działanie, w tym przypadku – 2 podniesione do szóstej potęgi. Teraz ćwiczenie: wykonaj za pomocą mikrokomputera ciąg działań od *2 do potęgi 0* do *2 do potęgi 16*, a ich wyniki zanotuj na kartce. Szereg potęgowy dwójki ma szczególne znaczenie w codziennej praktyce komputerowej.

Komputer wykonuje także działanie odwrotne do potęgowania: na klawiaturze nie ma jednak symbolu pierwiastkowania i trzeba go określić za pomocą kilku znaków. Wprowadź do mikrokomputera:

?SQR (4) [RETURN]

SQR *(liczba)* jest skrótem od angielskiego wyrażenia *SQUARE ROOT* i oznacza pierwiastek kwadratowy. Jeśli ktoś chciałby użyć innych funkcji matematycznych, niech zajrzy na stronę 125 podręcznika Commodore 64, gdzie zestawiono wszystkie funkcje obliczane przez mikrokomputer.

W tej chwili chcielibyśmy jednak zakończyć omawianie trybu bezpośredniego.

ETAPY PRACY MIKROKOMPUTERA

Zanim przejdziemy dalej, zastanów się, jak rozwiązuje się zadanie rachunkowe. Oczywiście zależy to od rodzaju problemu. Najczęściej analizuje się zadanie, wykonuje obliczenia i zapisuje wynik. Mikrokomputer postępuje identycznie, ale jego możliwości analizowania są ograniczone. Właściwym zadaniem mikrokomputera jest wykonywanie obliczeń i może on to czynić znacznie szybciej od człowieka. Daleko mu jednak do inteligencji, choćby sztucznej. Należy więc wykorzystać jego szybkość w liczeniu, a samemu zająć się analizą zadania. Pamiętaj, że mikrokomputer jest twoim niewolnikiem, ale

musisz dostosować się do jego możliwości. Komputer pracuje na tyle skutecznie, na ile dobry był jego programista. Rozpatrzmy teraz szczegółowo trzy zasadnicze etapy pracy komputera.

WPROWADZANIE

Na tym etapie, podobnie jak i przy wyprowadzaniu, następuje bezpośredni kontakt użytkownika z mikrokomputerem. Użytkownik dostarcza mikrokomputerowi dane, a także niezbędny plan działania – program. Powinieneś zawsze konstruować programy tak, aby ich użytkownik – także, gdy będzie to ktoś obcy – wiedział, jakich danych wejściowych oczekuje komputer.

PRZETWARZANIE

Przetwarzanie, czyli mówiąc trywialnie obliczanie, jest zadaniem nie tylko mikrokomputera, lecz także programisty. Komputer ma ściśle określone możliwości pracy i programista powinien formułować problemy tak, aby mikrokomputer był w stanie je *strawić*. Skuteczne przetwarzanie, czyli pełne wykorzystanie mikrokomputera, osiągniesz poznając dobrze jego możliwości. Nasz kurs powinien Ci w tym pomóc.

WYPROWADZANIE

Po wygodnym wprowadzeniu danych i sprawnym przetworzeniu należy oczekiwać wyprowadzenia wyniku w czytelnej i eleganckiej postaci. Warto pomęczyć się trochę przy programowaniu, aby zakończenie pracy odbyło się równie sprawnie, jak poprzednie etapy. Nieco więcej informacji o wprowadzaniu i wyprowadzaniu będzie też w następnym odcinku.

INSTRUKCJA PRINT

Poznałeś już najprostszą formę przetwarzania – wykonywanie operacji arytmetycznych. Mimochodem zaznajomiłeś się też z instrukcją wyprowadzania z mikrokomputera, i to z użyciem pewnego triku, a mianowicie formy skróconej. Znak zapytania ? jest bowiem skróconym zapisem instrukcji *PRINT*. Spróbuj jeszcze raz przećwiczyć z komputerem podane wcześniej przykłady, zastępując znak ? słowem *PRINT*. Zobaczysz, że prowadzi to do identycznych rezultatów.

DLA UŻYTKOWNIKÓW ZX SPECTRUM

W ZX Spectrum w zgłoszeniu gotowości nie występuje słowo *READY*. Po wykonaniu polecenia, przy realizacji którego nie było problemów, wypisywany jest komunikat: *O.K.*

BASIC w ZX Spectrum nie dopuszcza stosowania skróconej formy instrukcji *PRINT*. Zamiast:

?2+2 (klawisz RETURN)

musimy napisać:

PRINT 2+2 (klawisz ENTER)

Nieco inny jest też sposób wprowadzania instrukcji. Aby umożliwić podawanie całych instrukcji za pomocą jednego klawisza, przewidziano specjalne tryby wprowadzania sygnalizowane wyświetlaniem różnych liter, spełniających funkcje kursora. Gdy komputer oczekuje wprowadzenia komendy, wtedy wyświetlana jest litera *K*. Tryb ten pozwala wprowadzać komendy wypisane białym kolorem na klawiszach. Aby wprowadzić komendę wypisaną zielonym lub czerwonym kolorem na obudowie komputera (nie na klawiszach) należy równocześnie przycisnąć klawisze z napisami:

CAPS SHIFT oraz **SYMBOL SHIFT**

Jako kursor pojawi się wtedy litera *E*. Dla wprowadzenia *zielonej* komendy wystarczy teraz przycisnąć klawisz znajdujący się pod napisem. Dla wprowadzenia *czerwonej* komendy należy nacisnąć ponownie klawisz z napisem *SYMBOL SHIFT* i równocześnie klawisz znajdujący się nad tym napisem.

Po wprowadzeniu komendy komputer przechodzi do wprowadzania w trybie sygnalizowanym literą *L*. Klawisze reprezentują teraz umieszczone na nich litery bądź cyfry. Znaki wypisane na klawiszach kolorem czerwonym można wprowadzać przyciskając równocześnie z wybranym klawiszem *SYMBOL SHIFT*.

Liczby wyprowadzane jako wynik operacji zrealizowanych przez ZX Spectrum są poprzedzane przez znak tylko wtedy, gdy jest to minus. Przed liczbami dodatnimi nie jest wyprowadzana spacja (odstęp).

Do sterowania kursorem stosowane są klawisze *5, 6, 7* i *8*, nad którymi umieszczono odpowiednie strzałki. W trybie edycji (poprawiania błędów) kursor można przesuwać jedynie w poziomie i to nie dalej niż dotychczas wprowadzony tekst. W tym celu należy przycisnąć klawisz *CAPS SHIFT* i klawisz *5* (kursor w lewo) lub klawisz *8* (kursor w prawo).

Kasowanie odbywa się przez ustawienie kursora za zbędnym znakiem i przyciśnięcie klawiszy: *CAPS SHIFT* i *0 (DELETE)*. Wstawianie dodatkowych znaków wymaga jedynie ustawienia kursora w żądanym miejscu. Wpisywanie rozsuwa dotychczasowe znaki. Dla zastąpienia jednego zapisu innym, niezbędne jest wykasowanie niepożądanych znaków i wpisanie nowych.

SŁOWNICZEK

Mówiąc o programowaniu powinniśmy posługiwać się prawidłową wymową słów wywodzących się z języków obcych (najczęściej angielskiego). Pomocny powinien być w tym umieszczany przy każdym odcinku słowniczek nowych terminów.

READY *(redi)* – gotów
RETURN *(ritern)* – wróć
ENTER *(enter)* – wprowadź
SHIFT *(szift)* – przesuń
INSERT *(insert)* – wstaw
DELETE *(dilijt)* – usuń
SQUARE *(skłer)* – kwadratowy
ROOT *(ruut)* – pierwiastek
PRINT *(print)* – drukuj

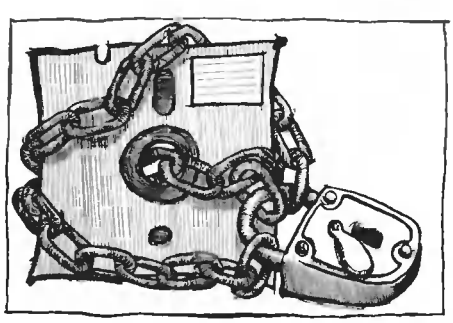
micro

WINCHESTER

O tym, jak szybko postępuje miniaturyzacja pamięci dyskowych, świadczy wprowadzona ostatnio na rynek przez francuską firmę Bull, jednostka pamięci D 585 na minidyskach typu Winchester. Przy średnicy zaledwie 5 1/4 cala (ok. 130 mm) pamięć ta ma niewiarygodną wprost pojemność 85 MB. Pojemność tę uzyskano głównie dzięki zagęszczeniu ścieżek do poziomu 1047 tpi (ang. tracks per inch – ścieżek na cal). Spośród istotnych parametrów tej pamięci należy wymienić stosunkowo krótki czas dostępu, wynoszący 30 ms, oraz standardowy interfejs St/412/St 506. Pewność zapisu danych zapewniają: automatyczna blokada mechanizmu ustawiania głowic oraz optymalne tłumienie drgań głowic. Obszary zastosowań to praca wielozadaniowa i wielodostępna oraz profesjonalna grafika przy użyciu sprzętu mikrokomputerowego. Zaskakuje również cena tej pamięci, która wynosi ok. 1600 dolarów.

ZABEZPIECZENIE PROGRAMU GEM

Próbując obdarzyć IBM PC możliwościami graficznymi, porównywalnymi do Macintosh'a, firma Digital Research (ta od CP/M) opracowała program narzędziowy o nazwie GEM. Zapewne w przyszłości poświęcimy mu trochę miejsca w MIKROKLANIE. Na razie ostrzeżenie dla programowych piratów. Program GEM ma zabezpieczenie pozwalające stwierdzić, czy pracuje na oryginalnym IBM PC, czy też na jego odpowiedniku (np. Olivetii M24, Eagle lub Zenith 150). Jeżeli nie dysponujemy odpowiednio przystosowaną wersją, próba wykorzystania GEM zakończy się komunikatem: *This version of GEM will not run on this computer* (ang.) *ta wersja GEM nie będzie działała na tym komputerze* i powrotem do systemu operacyjnego. Zabezpieczenie to zapewne można jakoś obejść, ale na razie nie rozpowszechniła się jeszcze informacja, jak tego dokonać. Firmy produkujące oprogramowanie coraz częściej zabezpieczają je przed nieautoryzowanym kopiowaniem, mimo że w większości krajów zachodnich uchwalono już ustawy *antypirackie*.



PIERWSZE KROKI...

Zdecydowaliście się zastosować mikrokomputer do zautomatyzowania prac biurowych w Waszym przedsiębiorstwie lub instytucji. Ma on pomóc w zarządzaniu, planowaniu i sprawozdawczości, gospodarce materiałowej, obsłudze odbiorców i przypominać o upływających terminach. Po długim namyśle wybraliście wreszcie typ mikrokomputera oraz odpowiednie dla Waszych potrzeb oprogramowanie. Przeczytajcie więc, w jaki sposób stawiać pierwsze kroki z mikrokomputerem.

Zarezerwujcie sobie jeden weekend, aby opracować dokładny plan postępowania. Przestawienie się na zastosowanie w przedsiębiorstwie lub instytucji mikrokomputera wymaga racjonalnego zorganizowania prac biurowych. Przemyślcie spokojnie i starannie wszystkie związane z tym aspekty. Dzięki temu unikniecie błędów z powodu niewłaściwej organizacji przedsięwzięcia.

Wprowadzenie techniki mikrokomputerowej obejmuje cztery fazy:

- planowania,
- wprowadzania,
- przestawiania,
- eksploatacji.

W fazie planowania należy przemyśleć i wykonać następujące czynności:
– opracować harmonogram przechodzenia na technikę komputerową
– zestawić i uporządkować wszystkie podstawowe informacje, jakie muszą być wprowadzone do komputera jako dane początkowe
– ustalić właściwy sposób kodowania w komputerze informacji, np. o odbiorcach (dostawcach) i towarach (materiałach),
– określić format i treść informacji wyjściowych, a więc zaprojektować potrzebne zestawienia i formularze, oraz ustalić czas, w jakim muszą być one wydrukowane,
– przeprowadzić szkolenie w posługiwaniu się wybranym pakietem programów.

Istotą takiego szkolenia jest bezpośredni kontakt z autorami oprogramowania. Powinni oni przekazać Wam informacje i wskazówki, których niestety nie znajdziecie w podręczniku. Program bardzo rzadko można od razu i bez kłopotów uruchomić. Dlatego rozmowy z fachowcem oszczędzają Wam czas uczenia się na własnych błędach.

W fazie wprowadzania tworzone są podstawy do pracy z mikrokomputerem. Zaczynajcie wprowadzać dane. Także i tu zaplanujcie odpowiednio dużo czasu. Rozsądniej jest więc pracować wolniej, unikając przez to błędów, które potem wymagałyby wprowadzania kosztownych korekt.

W każdym razie, aby np. zautomatyzować operacje sprzedaży towarów, musicie wprowadzić wszystkie dane o odbiorcach, towarach oraz dotychczas otwartych kontach księgowych. Przy dobrej organizacji pracy można w ciągu godziny wprowadzić np. dane o 50 odbiorcach. Konta księgowe wprowadza się wprawdzie szybciej, ale dane o towarach zajmą Wam nieco więcej czasu. Za minimalny czas realizacji całej fazy wprowadzania można uznać jeden miesiąc. Jeśli przetwarzanie danych na komputerze jest dla Was kompletną nowością, powinniście oczywiście zaplanować na tę fazę dłuższy okres. Dotyczy to także przypadków, gdy liczba danych o odbiorcach i towarach jest bardzo duża, lub gdy codziennie na wprowadzanie możecie przeznaczyć jedynie ograniczoną ilość czasu.

W fazie przestawiania zaczynacie pracować z mikrokomputerem i programem, na początku w formie ćwiczeń. Aby przeprowadzić wszystkie prace szybko i pewnie, zaleca się wykonywać je dwutorowo. Należy więc nadal pracować w sposób dotychczasowy, a równolegle do tego wykonywać wszystkie prace na mikrokomputerze. Faza ta powinna trwać od kilku dni do kilku tygodni.

Także tutaj obowiązuje zasada działania bez pośpiechu. Początkowy wzrost kosztów, spowodowany dublowaniem pracy, zostanie szybko zrekompensowany oszczędnościami wynikającymi z szybkiej i niezawodnej pracy mikrokomputera. I jeszcze jedna rada dotycząca fazy przestawiania: gdy wprowadzicie już wszystkie dane, należy dyskietki skopiować, aby w czasie prób i szkolenia informacje nie zostały zniszczone lub zmienione wskutek popełnienia błędu.

Obecnie dostępne są stacje dysków elastycznych (dyskietek) o pojemności od 180 tys. do 1.5 mln bajtów (znaków) oraz stacje dysków sztywnych o pojemności od 5 do 40 mln bajtów. Wybór właściwej pojemności zależy przede wszystkim od ilości informacji, które trzeba umieścić w pamięci dyskowej. Pojemność tę musicie sami oszacować, przewidując oprócz tego odpowiednią rezerwę, wynikającą z rozwoju Waszego przedsiębiorstwa, a więc wzrostu liczby zawieranych transakcji.

Niezbędna ilość miejsca na dyskietce lub dysku sztywnym zależy w znacznym stopniu od oprogramowania. Dokładne dane na ten temat powinniście znaleźć w instrukcji producenta. Nie poprzestajcie nigdy na przybliżonych oszacowaniach. Uwzględnijcie również fakt, że większość programów potrzebuje do rejestracji danych tylko jednego z dwóch dostępnych w mikrokomputerze napędów dysków. Drugi napęd jest zarezerwowany dla dysków z zapisem programu.

Dyskietki (dyski elastyczne) jako pamięć masowa mogą zaspokoić potrzeby tylko małej firmy. Duże przedsiębiorstwo, mające wielu odbiorców i duży asortyment towarów, wymaga zastosowania dysków sztywnych.

Wasz mikrokomputer powinien mieć możliwość zapisania na jednej dyskietce informacji o pojemności co najmniej 500 000 znaków – w przeciwnym wypadku Wasza praca przypominać będzie czynności disk-jockey'a.

Złożone programy, do których należą programy do zarządzania przedsiębiorstwem, zajmują więcej niż 500 tys. bajtów. Zaleca się, aby programy znajdowały się w całości na jednej dyskietce. Zmiana dyskietki w czasie pracy jest niewygodna dla użytkownika, a co gorsza, zabiera cenny czas i może być źródłem błędów.

SZTYWNY DYSK JEST LEPSZY

Przy podejmowaniu decyzji o przestawieniu się na mikrokomputer należy rozróżnić szybkość samego mikrokomputera oraz szybkość, jaką można osiągnąć przy obsłudze maksymalnej liczby zadań przypadających ne jeden dzień pracy. Najczęściej kupowane mikrokomputery i programy nie pozwalają ani na równoczesną pracę kilku osób na jednej maszynie, ani na jednoczesną obsługę kilku zadań. Na ogół nie jest więc możliwe równoczesne przeprowadzanie obliczeń i drukowanie wyników.

Ponieważ dla programów zarządzania przedsiębiorstwem najważniejszą sprawą jest bezpieczeństwo danych, większość czasu mikrokomputera przeznaczona jest na przesyłanie danych pomiędzy mikrokomputerem a dyskiem (elastycznym lub sztywnym). Zapis lub odczyt danych o odbiorcy, z dysku elastycznego, odbywa się na ogół szybko: przeciętnie trwa to jedną sekundę. Przy bardzo dużych ilościach danych zadanie to trwa nieco dłużej. Więcej czasu zabiera zapis lub odczyt transakcji. Na zapamiętanie na dyskietce rachunku obejmującego 20 pozycji towarowych potrzeba około jednej minuty. Jeśli rachunek taki zostanie zapisany na sztywnym dysku, to dzięki większej szybkości transmisji danych, realizacja takiego zadania ulegnie znacznemu przyspieszeniu: cała operacja zakończy się po 15 sekundach. Z tych powodów przy większej liczbie transakcji, przypadających na jeden dzień, właściwym rozwiązaniem może okazać się zastosowanie sztywnego dysku.

NAJWAŻNIEJSZA JEST ORGANIZACJA

Na początku pracy z komputerem odniesiecie z pewnością wrażenie, że informacje zapisane na dysku są trudniej dostępne. Jeśli jesteście przyzwyczajeni do szufladek z fiszkami czy segregatorów z kartami, doznacie na początku niemiłego uczucia,

gdy wprowadzane dane „znikają" na dysku i wydaje się, że nigdy ich już nie odnajdziecie. Będzie się Wam wydawać, że przy pracy z mikrokomputerem wszystko jest mniej przejrzyste i okropnie kłopotliwe. Być może obecnie dysponujecie fiszką zawierającą wszystkie istotne informacje o odbiorcy, po którą wystarczy sięgnąć jednym ruchem. Z komputerem trwa to niemal wieczność, zanim zostanie wprowadzony program i poszukiwana informacja pojawi się na ekranie. A przecież chcielibyście jeszcze wyświetloną zawartość ekranu wypisać na drukarce... ile zajmie to czasu?

Już wkrótce przyzwyczaicie się do odmiennej organizacji Waszej pracy z mikrokomputerem i zapomnicie o tych początkowych problemach. Nadal najważniejsze informacje będziecie mieli zawsze na kartce pod ręką, ale zamiast fiszek będzie to przejrzyste zestawienie, i co ważniejsze – bez większego wysiłku zawsze aktualne. Odpadnie stała troska o wiele codziennych drobnych spraw, takich jak na przykład pilnowanie terminów wysyłki monitów do odbiorców, zawierających przypomnienie o upływającym terminie płatności. Te czasochłonne i szarpiące nerwy czynności wykona Wasz mikrokomputer, który samodzielnie wydrukuje wspomniane przypomnienia.

W codziennej Waszej pracy najważniejsze mogą być, przykładowo, następujące zestawienia przygotowywane automatycznie przez komputer:

- lista odbiorców z dokładnymi adresami oraz numerami telefonów i teleksów,
- spis inwentaryzacyjny, umożliwiający szybki wgląd w stan zapasu poszczególnych towarów,
- zestawienie faktur, z którego wynika ile zarobiliście na każdej transakcji,
- zestawienie nieuregulowanych należności, wykazujące opieszałych płatników,
- zestawienie wpływów pieniężnych – eliminujące wertowanie wyciągów bankowych.

Dzięki takim informacjom najważniejsze dane o stanie i wynikach Waszej działalności stają się bardziej uporządkowane, przejrzyste i łatwiej dostępne. Równie logicznie uporządkowana i przemyślana powinna być organizacja Waszych danych w mikrokomputerze.

DANE O PRZEDSIĘBIORSTWIE

Musicie dobrze przemyśleć sposób, w jaki zamierzacie ponumerować wystawiane faktury (potwierdzenia zamówień, dostaw i zapłat, noty kredytowe). Jeśli zaczniecie od numeru 1, to przy sześciocyfrowych numerach dokumentów pierwszy z nich otrzyma numer 000001. Jeśli przyzwyczailiście się w numerach faktur ujmować również datę, to operację taką można robić także przy użyciu mikrokomputera, np. nr 387/85 otrzyma w komputerze nr 850367, a nr 20 wystawiona w styczniu 85 – nr 850120.

Ważne jest, abyście utrwalili w swej świadomości, że program sam ustala kolejne numery faktur.

Powinniście dokładnie egzekwować uzgodnione z Waszymi odbiorcami warunki płatności – i w tym właśnie celu działa wspomniany system przypominania. Można również wprowadzić różne zmniejszenia należności (rabaty) oraz prolongaty terminu płatności.

DANE O ADRESACH

Do adresów Waszych odbiorców (lub dostawców), umieszczonych w pamięci mikrokomputera, traficie za pomocą tzw. klucza. Najlepszym kluczem jest najczęściej przyporządkowany każdemu kontrahentowi jego numer, obejmujący od 8 do 10 cyfr. Dobre programy pozwalają również na znalezienie adresu poprzez podanie nazwiska (firmy) kontrahenta, niezależnie od podania jego numeru. Dane o adresach będą w każdym razie uporządkowane w mikrokomputerze według numerów kontrahentów. Dlatego wybór sposobu ich numerowania (nadawania symboli) jest sprawą bardzo ważną. Symbol taki zwykle oprócz cyfr może zawierać także litery i inne znaki pisarskie (tzw. znaki specjalne). Sposób kodowania musi być logiczny, umożliwiający szybkie odnajdywanie osób lub firm.

Jeżeli najważniejszym kryterium przy odnajdywaniu jest nazwisko, to powinno ono wchodzić w skład symbolu kontrahenta. Jeżeli ważniejszy jest adres, symbol powinien zawierać kod pocztowy. Dzięki temu można zaoszczędzić wiele pracy, np. przy sporządzaniu list adresowych wymagających uprzedniego sortowania.

Przykład: Pan dr E. Neumann z Salzburga może być zakodowany w komputerze na różne sposoby[1]

6000neuman
6000neu01
60n01
60neum

Aby znaleźć optymalny sposób kodowania symboli kontrahentów, należy przestrzegać następujących zasad:

- symbole powinny być maksymalnie krótkie, ponieważ znacznie przyspiesza to dostęp do nich,
- powinny być dostatecznie długie, aby jednoznacznie rozróżniać poszczególnych kontrahentów i tym samym zmniejszać możliwość pomyłek przy wyszukiwaniu informacji,
- powinny zawierać przynajmniej część nazwiska lub adresu, aby uniknąć ciągłego przeglądania pełnej listy kontrahentów,
- powinny być jednolite – np. należy używać tylko małych liter, gdyż upraszcza to wprowadzanie danych,
- nie powinny zawierać odstępów (spacji), ponieważ trudno jest zapamiętać miejsce, w którym zostały one umieszczone,
- sortowanie powinno odbywać się według początkowych znaków symbolu: znaki te powinny być kodowane według kryterium uwzględniającego specyfikę Waszego przedsiębiorstwa.

Powinniście zwrócić uwagę również na to, aby przy informacji o adresie pozostało jeszcze miejsce na dodatkowe informacje o kontrahencie. W miejsce to będzie można wówczas wpisać, na przykład, skrót symbolu branży.

[1] Salzburg ma numer kodu pocztowego 6000 (przyp. red.)

GOSPODARKA MAGAZYNOWA

Także w tej dziedzinie najważniejszą sprawą jest wybór sposobu kodowania symboli towarów (materiałów). Symbole te składają się najczęściej z 6 do 8 znaków – cyfr, liter lub innych znaków pisarskich. Prawdopodobnie przy ustalaniu symboli będziecie opierać się możliwie najdokładniej na Waszym dotychczasowym systemie kodowania. Zastanówcie się jednak, czy nie byłoby rozsądnie kodować towary wyłącznie za pomocą cyfr. Uprościłoby to ich podział na grupy.

Jeśli początkowe dwie lub trzy cyfry symbolu towaru zarezerwujecie na oznaczenie grupy towarów, to Wasze dane magazynowe będą automatycznie uporządkowane zgodnie z rosnącymi numerami tych grup. Rozwiązanie takie zapewnia łatwość i przejrzystość prowadzenia ewidencji magazynowej.

Przykład: symbol towaru 50001 zawiera następujące informacje:

- grupa towarów nr 50
- pierwszy towar w tej grupie (001).

Również tu mają zastosowanie uwagi podane przy omawianiu zasad budowy symboli kontrahentów.

Następną istotną sprawą przy ustalaniu treści danych podstawowych jest udzielanie rabatu. Należy tu uwzględnić następujące aspekty:

- obniżki udzielane określonym grupom odbiorców
- stopniowanie obniżek dla poszczególnych klientów
- obniżki zależne od ilości i rodzaju kupowanego towaru
- obniżki odnoszące się do określonych grup towarów
- stopniowanie cen.

Od przyjętej w Waszej firmie polityki sprzedaży zależy, jakie obniżki stosujecie. W każdym razie należy unikać stosowania zagmatwanych metod obliczania obniżek, gdyż wtedy łatwo może się zdarzyć, że zaczniecie sprzedawać towary poniżej ich kosztu własnego.

FORMULARZE ZWIĄZANE Z FAKTUROWANIEM

Są to: potwierdzenia zamówień oraz dostawy, rachunki, noty kredytowe.

Program powinien umożliwiać ukształtowanie formy drukowanych formularzy zgodnie z Waszymi potrzebami. Powinniście ustalić sami, co i na jakiej pozycji me się znajdować w Waszym formularzu. W dobrych, wygodnych dla użytkownika programach, służy do tego celu tzw. generator list i formularzy. Oszczędza on Wasz czas i pieniądze. Dostosowanie programu przez producenta oprogramowania ściśle do Waszych potrzeb byłoby bardzo kosztowną i czasochłonną procedurą, a przy tym nie mielibyście możliwości ewentualnego późniejszego korygowania ustalonych formatów wydruków. Może się to okazać niezbędne dopiero po pewnym czasie eksploatacji systemu.

WYDRUK CIĄGŁY CZY NA ODDZIELNYCH KARTKACH?

Od początku pracy musicie zdecydować się, w jakiej formie Wasze rachunki i inne dokumenty mają być wypisywane na drukarce.

Możecie stosować w tym celu długie wstęgi papieru, zaopatrzone na krawędziach bocznych w rząd dziurek prowadzących papier w drukarce. Poszczególne dokumenty drukujecie wtedy na wydzielonym odcinku wstęgi. Jeśli zastosujecie czysty papier, na każdym formularzu trzeba będzie wydrukować firmowy nagłówek, co oczywiście zabiera cenny czas. Bardziej eleganckie rozwiązanie polega na użyciu folii z dziurkami prowadzącymi, do której przytwierdzone są arkusze papieru z wydrukowanym firmowym nagłówkiem. Po zadrukowaniu arkusze te odrywa się. Rozwiązanie to umożliwia również drukowanie przez kalkę potrzebnej liczby kopii dokumentu.

Innym rozwiązaniem jest użycie pojedynczych arkuszy papieru, wkręcanych do drukarki ręcznie lub przez mechanizm wprowadzający. Ręczne wkręcanie papieru opłaca się tylko przy bardzo małej dziennej liczbie wydruków. W najprostszym przypadku używacie znormalizowanych arkuszy papieru z nadrukiem firmowym.

Nie łudźcie się, że dzięki opisanym tu przedsięwzięciom wszystkie prace z miejsca zaczną przebiegać znacznie szybciej i efektywniej. Na początku będzie wprost przeciwnie. Musicie się nastawić na to, że przez okres przejściowy, trwający przynajmniej trzy miesiące, w czasie którego dokładnie zapoznacie się z mikrokomputerem i programem, będziecie mieli więcej pracy niż dotąd.

Później posługiwanie się mikrokomputerem stanie się czynnością rutynową. Będziecie wykonywać Wasze prace szybciej, a przede wszystkim będziecie mogli wykorzystać różnorodne i bardzo dokładne informacje o biegu Waszej działalności. Wynagrodzi to z nawiązką Wasz trud.

micro

TRZY I PÓŁCALOWE NIEMOWLĘ

Według szacunkowych ocen ekspertów w 1984 roku sprzedano na świecie ponad 400 milionów dyskietek.

Gdy w 1970 roku pojawiła się na rynku światowym pierwsza dyskietka, nie wzbudziła większego zainteresowania wśród użytkowników pamięci zewnętrznych. Był to krążek tworzywa pokryty warstwą magnetyczną, zaspawany w elastycznej kopercie z tworzywa sztucznego w kształcie kwadratu o boku 8 cali (200 mm). Musiały upłynąć trzy lata, zanim przekonano się o przydatności dyskietki do rejestracji i wymiany informacji. Początkowo obszar zastosowań dyskietki (nazwanej też dyskiem elastycznym lub z angielska floppy-dyskiem) był ograniczony. Wkrótce jednak obszar ten znacznie się rozszerzył.

Cztery istotne cechy zapewniły dyskietce przewagę nad poprzednio stosowanymi nośnikami informacji – taśmami i kartami dziurkowanymi oraz kasetowymi taśmami magnetycznymi:

- Stosunkowo duża pojemność – na pierwszych dyskietkach można było zapamiętać 250 tys. znaków, co stanowiło równowartość 140-150 stron maszynopisu lub ok. 3 tys. kart dziurkowanych. A przecież 8-calowa dyskietka była niewiele większa od powierzchni dwóch kart dziurkowanych!
- Krótki czas dostępu do danych, zapamiętywanych w bezpośrednio adresowanych sektorach, rozmieszczonych na koncentrycznych ścieżkach. Czas ten dla mikrodyskietki wynosi kilka mikrosekund, podczas gdy dla kart dziurkowanych – kilka sekund, a dla taśmy magnetycznej – kilka minut.
- Wygoda użytkowania – manipulowanie danymi zapisanymi na dyskietce odbywa się za pośrednictwem klawiatury, a ich treść może być wyświetlana na ekranie monitora.
- Niska cena.

Dzisiejsze dyskietki mają różne rozmiary. Najmniejsze – mieszczące się w kieszeni kamizelki (nie polecamy jednak takiego sposobu przechowywania) – mają wymiary 80 × 100 mm. Stosowane są dwa różne sposoby zapisu, elastyczne albo sztywne koperty, sektory o różnej długości oraz różne metody organizacji danych. Nawet materiały magnetyczne oraz nowe technologie powlekania pozwalają spodziewać się znacznego zwiększenia gęstości zapisu jeszcze w bieżącym dziesięcioleciu.

ZMNIEJSZANIE FORMATÓW

Pierwotna wersja 8-calowa pozostała w zasadzie nośnikiem informacji przy eksploatacji dużych komputerów. Obok dyskietek, zapisywanych jednostronnie i z pojedynczą gęstością, pojawiły się zapisywane dwustronnie i (lub) z podwójną gęstością. Wynikłe z tego zróżnicowania rozwiązań odmiany dyskietek zostały już znormalizowane.

W 1977 r. pojawiła się minidyskietka – dyskietka zmniejszona do średnicy 5,25 cala. Znalazła ona zastosowanie głównie w systemach zdecentralizowanych. Zmniejszone zostały jedynie wymiary krążka pokrytego warstwą magnetyczną, natomiast pozostałe parametry dyskietki pozostały bez zmian. Dopiero w następnych wersjach podwojono gęstość oraz liczbę stron zapisu,

Parametry dyskietek

Parametr	Typ dyskietki (w calach)				
	8	5.25	4	3.5	3.1
Długość (mm)	200	130	102	94	100
Szerokość (mm)	200	130	102	90	80
Osłona	elastyczna	elastyczna	sztywna	sztywna	sztywna
Gęstość zapisu (bitów/mm)	257	218	270	343	352
Gęstość ścieżek (ścieżek/mm)	1,9	3,8	2,7	5,3	3,9
Liczba zapisywanych stron	2	2	1	2	2
Pojemność (megabajtów)	1,6	1,0	0,358	1,0	0,5

a także gęstość rozmieszczenia ścieżek, tworząc w ten sposób całą rodzinę dyskietek pięć i ćwierćcalowych. Ich pojemność waha się od 0,125 do 1 mln znaków.

Czy można dalej zmniejszać wymiary dyskietki? Odpowiedź twierdzącą przyniosło pojawienie się mikrodyskietek o średnicy mniejszej niż 5 cali (ang. *sub five* – poniżej pięciu). Różnią się one od wersji 8- lub 5,25-calowej sztywną osłoną z automatycznie uruchamianym zamknięciem otworu na głowicę oraz nowym rodzajem powłoki magnetycznej, umożliwiającej zwiększenie gęstości zapisu. Popularna wersja mikrodyskietki ma osłonę o wymiarach 3,5 na 3,7 cala, a pojemność informacyjną taką samą, jak prawie dwukrotnie większa wymiarem minidyskietka 5,25-calowa. Osłona i wspomniany mechanizm zamykający skuteczniej niż dotąd zabezpieczają dyskietkę przed uszkodzeniem i zanieczyszczeniem. Znajdzie ona zapewne zastosowanie w komputerach osobistych nowej generacji.

Obecnie o wejście na rynek współzawodniczą cztery rodzaje mikrodyskietek. Różnią się one znacznie – co jest istotnym novum – swoimi zewnętrznymi cechami konstrukcyjnymi. Nie wiadomo jeszcze, które rozwiązanie wybiorą użytkownicy. W tabeli porównano podstawowe parametry trzech rodzajów mikrodyskietek oraz dyskietek 8- i 5,25-calowej. Producenci zapowiadają dalszy rozwój ich konstrukcji. Można przypuszczać, że prace badawcze w tym kierunku wyszły już prawdopodobnie poza fazę koncepcji.

KONIECZNOŚĆ NORMALIZACJI

Zastosowanie dyskietek w wielu różnych dziedzinach spowodowało niezależne i nieskoordynowane (zarówno w skali poszczególnych krajów, jak i w skali międzynarodowej) opracowanie wielu różnych rozwiązań. Dość powiedzieć, że największy europejski producent nośników informacji wytwarza ok. 400 różnych odmian dyskietek, na ogół tylko nieznacznie różniących się między sobą. Sytuacja taka jest bardzo niekorzystna zarówno dla producentów, jak i dla użytkowników.

PERSPEKTYWY DALSZEGO ROZWOJU

Przedstawiciele przemysłu są zgodni, że przy pełnym wykorzystaniu współczesnej technologii produkcji sprzętu i nośników, dyskietki mogą całkowicie wyprzeć małe pamięci na dyskach niewymiennych o pojemności 10 megabajtów, a nawet większych. Należy jednak napierw rozwiązać problemy zabezpieczenia danych oraz zmniejszyć koszty wytwarzania nowych wyrobów. O trudnościach, na jakie może natrafić taki ambitny program, może świadczyć historia całkowicie dotąd nieudanych prób skonstruowania kasety dla dysków typu Winchester.

Na razie nie widać perspektyw dalszego zmniejszania wymiarów dyskietek. Wątpliwe jest zmniejszanie średnicy poniżej 3,5 cala. Dyskietki o średnicy mniejszej niż 2 cale są już tak małe, że trudno nimi operować i łatwo je zgubić. Zdaniem fizyków i chemików długo jeszcze nie uda się wykorzystać pełnych możliwości materiałów magnetycznych.

Wróćmy do chwili obecnej. Oferowane są napędy pozwalające sformatować dyskietkę 5,25-calową, zgodnie z normą dla dyskietek 8-calowych o pojemności 1,6 megabajta. Zapowiedziano ostatnio dostawy napędów dla dyskietek 5,25 cala o pojemności ok. 2 megabajtów.

W obecnym dziesięcioleciu w rozwoju mikrokomputerów zapewne powtórzy się historia komputerów z lat sześćdziesiątych. Wielki wzrost wydajności mikroprocesorów i układów sterujących pracą kanałów komunikacyjnych, wraz ze wzrostem pojemności pamięci zewnętrznych, doprowadziły do wyposażenia mikrokomputerów w pamięci na dyskach niewymiennych. W pamięciach tych zabezpieczanie danych jest jednak procesem bardzo pracochłonnym. Dla dużych komputerów rozwiązano ten problem za pomocą strumieniowych jednostek pamięci taśmowej. Dla mikrokomputerów, jak dotąd, nie znaleziono zadowalających i jednocześnie tanich rozwiązań.

micro

SPECTRUM KONTRA... MACINTOSH

Przyrównywanie ZX Spectrum do komputera Macintosh ma taki sens, jak zestawienie malucha z Porsche. Przy gasnącym zainteresowaniu produktem firmy Sinclair, zaskakująca jest więc nowa oferta Oxford Computer Publishing. Program o nazwie „Art Studio" wprawił w zdumienie nawet koneserów grafiki komputerowej, którzy nie przypuszczali, że na mikroprocesorze 8-bitowym i to jeszcze pracującym w przestarzałym ZX Spectrum można osiągnąć podobne rezultaty. „Art Studio" (czyli: „Pracownia plastyczna") jest programem wzorowanym na sławnym MacPaint napisanym dla Macintosha i podobno niewiele mu ustępuje. Wykorzystano w nim oczywiście technikę okienek (podział ekranu na części, w których wyświetlane są niezależne od siebie obrazy) oraz ideogramy (ang. icons) zastępujące napisy. Obsługi programu nie trzeba się uczyć, gdyż użytkownik zawsze może sięgać do tzw. menu, czyli spisu możliwych operacji.

„Art Studio" odróżnia się od MacPaint możliwością stosowania kolorów (Macintosh ma ekran monochromatyczny). Istnieje jednak poważna niedogodność, wynikająca z konstrukcji ZX Spectrum. W komórce o rozmiarach znaku nie mogą ze sobą sąsiadować więcej niż dwie barwy. Jeśli się o tym zapomni, cały czar pryska.

Z programem „Art Studio" można współpracować wykorzystując klawiaturę (najmniej wygodny sposób), drążek sterowniczy lub mysz firmy AMX.

DRUKARKA STRUMIENIOWA

Coraz więcej wskazuje na to, że elitarna jak dotąd firma HEWLETT-PACKARD stara się pozyskać dla swoich wyrobów szerokie grono użytkowników komputerów osobistych. Świadczy o tym między innymi wprowadzenie na rynek nowego modelu niedrogiej drukarki strumieniowej (piszącej strumyczkiem tuszu) o nazwie ThinkJet Printer. O zasadzie działania i wcześniejszych modelach drukarek strumieniowych firmy HP można było przeczytać w numerze 3, 1985 INFORMATYKI. Najnowsza wersja o symbolu HP 2225 D wyposażona jest w interfejs szeregowy RS-232 C i może współpracować zarówno z komputerami oferowanymi przez producenta, jak i IBM PC/AT/XT, Apple IIc i IIe oraz Macintosh. Poprzednie modele różniły się typem zastosowanego interfejsu: model A ma złącze HPIB (IEEE 488), model B – HP-IL a model C – interfejs równoległy Centronics.

JAK DBAĆ O DYSKIETKI?

Prawidłowe posługiwanie się dyskietkami pozwala uniknąć ryzyka utraty cennych, często niemożliwych do odtworzenia informacji. Warto więc opanować kilka podstawowych reguł postępowania zebranych w poniższym tekście.

Nagrywanie czy kasowanie piosenek na kasetach magnetofonowych nie stanowi dziś dla nikogo specjalnego problemu. Do taśm czy kaset zdążyliśmy się już przyzwyczaić i nie popełniamy większych błędów.

Postępowanie z nowymi magnetycznymi nośnikami informacji wymaga wyrobienia nowych nawyków, które należy przećwiczyć i utrwalić. Jest to szczególnie ważne, gdy mamy do czynienia z rejestracją informacji, których utrata mogłaby zdezorganizować pracę przedsiębiorstwa, a nawet doprowadzić je do bankructwa.

Najpowszechniej dziś stosowanym w mikrokomputerach nośnikiem informacji są dyski elastyczne. Spotyka się też dyski sztywne typu Winchester oraz mikrodyskietki zamknięte w kasecie. Znacznie rzadziej stosuje się obecnie taśmy magnetyczne.

Wszystkie wymienione rodzaje pamięci mają jedną cechę wspólną: dane są zapisywane i odczytywane z nośnika magnetycznego. Jednakże dyskietki są medium najbardziej podatnym na uszkodzenia, ze względu na bezpośredni kontakt z otoczeniem i tylko minimalne zabezpieczenie.

Dyskietka jest zbudowana z okrągłego kawałka plastikowej folii, pokrytej materiałem magnetycznym. Krążek ten jest zamknięty w kopercie z elastycznego, ale odpornego tworzywa. Duży, umieszczony po środku otwór, służy do ustawienia dyskietki na wałku mechanizmu napędowego. Brzegi otworu mogą być wzmocnione wąskim plastikowym pierścieniem, naklejonym bezpośrednio na warstwie magnetycznej. Nieco z boku znajduje się drugi, niewielki otwór, zwany otworem indeksowym. Odpowiada mu obustronne wycięcie w kopercie. Dzięki temu, w momencie przesunięcia się otworu indeksowego pod wycięciem, mikrokomputer może określić położenie obracającej się dyskietki względem głowicy zapisu-odczytu.

Trzeci otwór, o kształcie owalnym, jest wycięty tylko w kopercie. Jest to szczelina zapisu-odczytu, nad którą przesuwa się głowica pisząco-czytająca, po umieszczeniu dyskietki w urządzeniu napędowym. Okolicy szczeliny, a zwłaszcza widocznej przez nią warstwy magnetycznej – pokrywającej powierzchnię dyskietki, nie wolno nigdy dotykać palcami.

Nacięcie widoczne na jednej z krawędzi koperty, służy do blokowania zapisu. Można je zakleić, używając specjalnych etykiet dostarczanych wraz z dyskietką. W niektórych systemach blokadę zapisu powoduje nacięcie zaklejone, w innych zaś – nie zaklejone. Do zaklejania nacięć nie stosujcie nigdy taśm samoprzylepnych – przyklejają się one bowiem zbyt mocno, a z upływem czasu po prostu brudzą dyskietkę.

Dwa małe wgłębienia w krawędzi koperty, obok szczeliny zapisu-odczytu, zapewniają właściwe ustawienie dyskietki w urządzeniu napędowym.

Dyskietka, czyli plastikowa kwadratowa koperta zawierająca wewnątrz krążek z folii, jest dostarczana w papierowej lub tekturowej kopercie ochronnej. Wkłada się ją do urządzenia napędowego bezpośrednio po wyjęciu z papierowej koperty ochronnej. Do tej samej koperty dyskietka musi powrócić natychmiast po jej wyjęciu z urządzenia napędowego!

ZASADY POSTĘPOWANIA Z DYSKIETKAMI

Najważniejsza zasada brzmi: nigdy nie dotykać powierzchni warstwy magnetycznej! Najmniejsze jej zanieczyszczenie może spowodować utratę jej własności użytkowych. Jeśli w dotkniętym przez Was miejscu były zapisane, na przykład, obroty przedsiębiorstwa z ostatniego miesiąca – skutki takiej nieostrożności będą katastrofalne! A więc precz z paluchami!

Dyskietka jest wyrobem delikatnym. Wsuwając ją do urządzenia napędowego nie stosujcie nigdy siły! Jeżeli przy wkładaniu lub wyjmowaniu pojawią się trudności, należy sprawdzić, czy etykieta blokady zapisu dobrze przylega do koperty. Gdy mechanizm napędowy jest już w ruchu (zwykle świeci się wtedy lampka sygnalizacyjna) nie wolno ani wyjmować, ani wkładać dyskietki. Niepotrzebne już do dalszej pracy dyskietki należy wyjąć z mechanizmów napędowych przed ich włączeniem. Nie wolno też dopuścić, aby dyskietka znajdowała się w urządzeniu napędowym w momencie włączania lub wyłączania urządzenia z sieci.

Zasada druga: obchodźcie się z dyskietką tak delikatnie, jak z kobietą. Dlatego nigdy nie róbcie w dyskietkach dziur! Podobno byli już tacy, którzy przypinali dyskietki do ściany pinezką. Najlepszym miejscem ich przechowywania są pudełka z antystatycznego plastiku.

Dyskietkom szkodzą: wysoka temperatura, pola magnetyczne (telefony, głośniki, telewizory, odbiorniki radiowe), wilgoć, kurz, nawet dym z papierosa. Złośliwym, bo niepozornym źródłem zagrożenia może być namagnesowany śrubokręt lub długopis.

Zasada trzecia: chrońcie dyskietkę przed oddziaływaniami czynników zewnętrznych (łącznie z polami magnetycznymi i wyładowaniami elektrostatycznymi).

Jeśli chcecie mieć kłopoty – kupujcie dyskietki kiepskiej jakości. Nęcą one swą ceną – zaledwie jednego dolara za sztukę. Rozsądny człowiek niewątpliwie porówna ceny i walory użytkowe tanich dyskietek z analogicznymi cechami wyrobów dobrej jakości i nie da się skusić niską ceną. **Zasada czwarta:** kupujcie dyskietki wyłącznie dobrej jakości.

Zdarzają się przykre niespodzianki, np. rozlejecie kawę na zapisaną dyskietkę albo w sąsiedztwie uderzy piorun, co spowoduje całkowitą utratę informacji. **Zasada piąta:** zawsze należy zrobić kopię.

Tu kilka uwag na temat kopii. Możecie kopiować programy lub zbiory. Kopiowanie zakupionych programów jest obwarowane pewnymi przepisami, różnymi u różnych producentów. Zazwyczaj kupuje się licencję na wykorzystywanie programu. Oznacza to, że nie wolno robić kopii w celu jej sprzedaży lub odstąpienia innej osobie, lecz jedynie do wykorzystania na własny użytek. Istnieje nawet wymóg, aby sporządzać sobie kopię bezpieczeństwa i dostawca wyjaśnia istotę takiej procedury.

Inni producenci (choć jest to sporadyczny przypadek) stosują specjalne środki zabezpieczające przed nielegalnym powielaniem programu. Program można wprawdzie skopiować, ale nie daje się on uruchomić na mikrokomputerze. W takiej sytuacji trzeba dokładnie wypytać dostawcę lub producenta, co czynić w przypadku nieumyślnego zniszczenia zapisu. Nie można przecież oczekiwać całymi dniami na serwis, który ma dostarczyć nowy egzemplarz dyskietki z programem.

Ze zbiorami danych możecie robić co dusza zapragnie. Nie podlegają one umowom licencyjnym, są tworzone i wykorzystywane przez użytkownika. Zbiory te mogą być bardzo cenne – opracowano więc liczne zasady postępowania zapewniającego ich bezpieczeństwo.

Zbiory danych rozmnażają się! Pierwotny zbiór danych jest aktualizowany, to znaczy wprowadzone są do niego nowe dane, zaś stare są usuwane lub modyfikowane. Powstaje w ten sposób nowy zbiór, pochodzący od starego. Stary nazwijmy ojcem, zaś nowy – synem (choć równie dobrze można by je nazwać na przykład matką i córką).

Jeśli syn ulegnie zniszczeniu, zostanie ojciec, istnienie którego pozwala dość łatwo „zrobić" ponownie syna. Nie należy więc kasować zbioru nie mającego swego ojca.

Można tworzyć więcej generacji zbiorów. Z syna powstaje jego syn, z niego – syn syna syna, czyli prawnuczek pierwotnego zbioru. Po pewnym czasie pierwotne zbiory – pradziadkowie czy dziadkowie – będą tak zdezaktualizowane, że można je będzie skasować, robiąc na dyskietkach miejsce nowym zbiorom.

Jeśli cała rodzina zbiorów będzie przechowywana na tym samym nośniku (dysku elastycznym lub sztywnym, taśmie), ślepy los zawsze może ją zniszczyć. Aby się przed tym zabezpieczyć, należy poszczególne generacje zbiorów przechowywać na kilku nośnikach. Rzadko używane i szczególnie cenne zbiory należy przechowywać w odpowiednio bezpiecznym miejscu.

Reasumując, dochodzimy **do zasady piątej**: należy chronić nie tylko sam nośnik informacji, lecz również jego zawartość. Warto znać podstawowe zasady ochrony danych, ich naruszenie powoduje bowiem dotkliwe skutki finansowe. Trzeba więc przemyśleć organizację archiwum zbiorów.

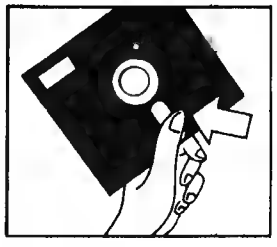
Po pierwsze, należy ujednolicić opisy na etykietach umieszczanych przez dostawcę na dyskietkach, aby nawet po dłuższym czasie szybko i prawidłowo zidentyfikować zawartość dyskietki. W opisach stosujcie jednoznaczne określenia, jednolite nazwy zbiorów oraz daty ich utworzenia i ostatniej aktualizacji.

Jeśli mimo wszystkich środków zabezpieczających powstała awaria – trzeba zachować spokój: nie zawsze zbiór jest nieodwracalnie stracony. Istnieją liczne programy przeznaczone do odtworzenia naruszonych zbiorów. Niestety nie zawsze się to udaje. Najskuteczniej można ratować zbiory wyłącznie tekstowe, w których powstałe luki dadzą się uzupełnić. Pewnych zbiorów danych – takich jak tabele lub zestawienia liczbowe – na ogół nie da się powtórnie ożywić.

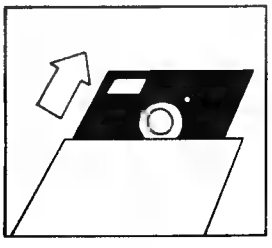
Mniej dramatyczna sytuacja powstaje po wydaniu nieprawidłowej komendy systemowi operacyjnemu (polecającej, na przykład, skasowanie zbioru). Istnieją programy, które niezawodnie odtworzą skasowany zbiór, jeśli tylko w tym samym miejscu na dysku nie została zapisana następna informacja.

Na zakończenie warto zapamiętać, że przy operowaniu zbiorami słuszna jest niewątpliwie stara maksyma lekarzy: *lepiej zapobiegać niż leczyć*.

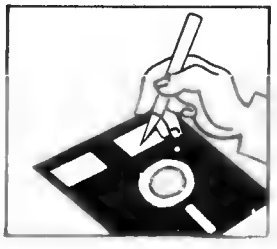
micro



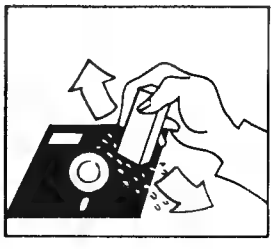
NIGDY nie dotykać powierzchni dyskietki



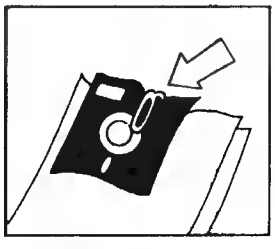
NIGDY potrzeby nie wyjmować dyskietki z osłony ochronnej



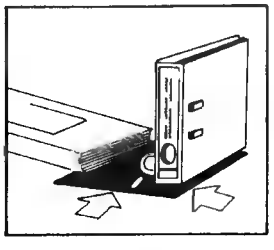
NIGDY nie opisywać dyskietki zaostrzonym ołówkiem lub długopisem



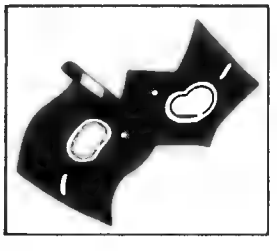
NIGDY nie wycierać gumką osłony ochronnej dyskietki



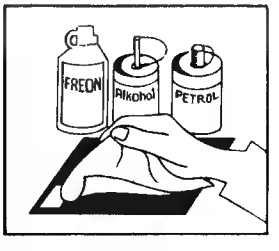
NIGDY nie mocować dyskietki taśmami gumowymi lub spinaczami biurowymi



NIGDY nie przyciskać dyskietki ciężkimi przedmiotami



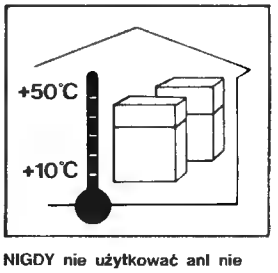
NIGDY nie wyginać tym bardziej zaginać dyskietki



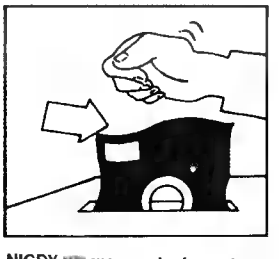
NIGDY nie czyścić dyskietki alkoholem, rozpuszczalnikiem ani freonem

NIGDY nie zbliżać dyskietki pobliże przedmiotów o własnościach magnetycznych

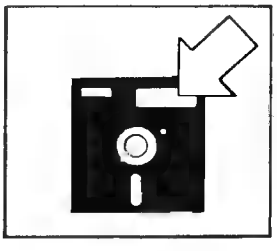
NIGDY nie wystawiać dyskietki działanie promieni słonecznych lub źródła ciepła



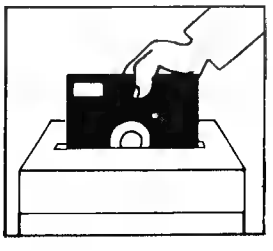
NIGDY nie użytkować ani nie przechowywać dyskietki temperaturach poniżej +10°C albo powyżej +50°C



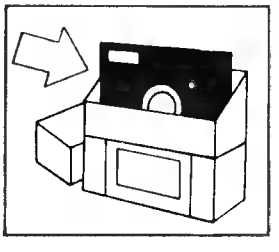
NIGDY wprowadzać na siłę dyskietki do urządzenia napędowego



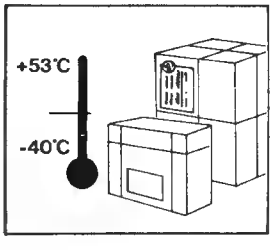
ZAWSZE opisywać dyskietkę miękkim pisakiem przewidzianym do tego celu polu etykiety



ZAWSZE ujmować dyskietkę za górną krawędź i uważnie wprowadzać urządzenia napędowego



ZAWSZE przechowywać dyskietkę pionowo w opakowaniu



ZAWSZE przesyłać dyskietkę tak, aby nie była narażona temperatury powyżej +53°C ani poniżej –40°C

NIGDY I ZAWSZE: wraz z upowszechnieniem komputerów osobistych popularność zdobyły również dyskietki. Aby w pełni zachować niezawodność i funkcjonalność dyskietek, należy przestrzegać podanych obok zasad postępowania.

BEZ STRESÓW

Obecnie, gdy mikrokomputery wyszły już z *lat dziecinnych*, można je spotkać w biurze, sklepie i magazynie. Za klawiaturą mikrokomputera zasiada sekretarka, sprzedawca, referent – czyli użytkownik bez wykształcenia w zakresie elektronicznego przetwarzania danych. Takiemu użytkownikowi nie wystarcza ergonomiczna forma sprzętu, chciałby *pogadać* ze swoim komputerkiem, a nie wyłącznie *wprowadzać* dane czy program.

Jeszcze niedawno uważano, że to człowiek musi dopasować się do maszyny. Dziś lansowany jest pogląd odwrotny. Ergonomia jest nauką zajmującą się dostosowywaniem środowiska pracy – maszyn i urządzeń – do potrzeb człowieka, z uwzględnieniem wielu czynników, jak np. oświetlenia, hałasu, klimatyzacji.

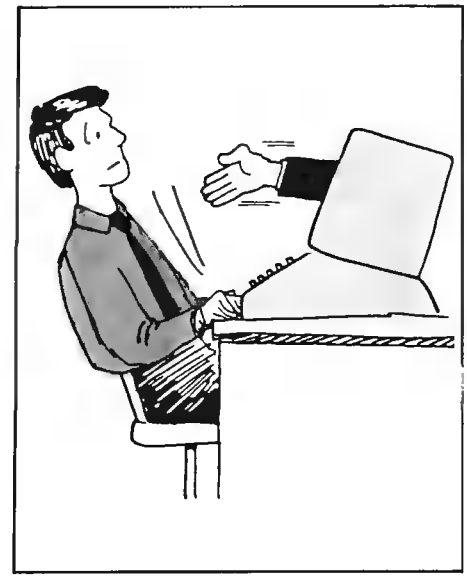
To, co teoretycznie jest oczywiste, dopiero niedawno dotarło do świadomości producentów komputerów. Początkowo sprzęt projektowano tak, aby był łatwy w montażu, a wygoda użytkownika pozostawała na dalszym planie. Skorzystała na tym branża mebli biurowych, sprzedając specjalne stoły, krzesła i stojaki, dopasowując toporny sprzęt do wymagań człowieka. Obecnie problem zniknął z źródła – producenci dbają o to, by ich sprzęt był ergonomiczny.

Nawet jeśli system komputerowy zewnętrznie spełnia podstawowe wymagania ergonomiczne, wcale nie znaczy to, że będzie wygodny w obsłudze. Bardzo dużo zależy bowiem od dostosowania oprogramowania do mentalności człowieka. Dziś współpraca z komputerem nie odbywa się już za pomocą formularzy i kart perforowanych. Użytkownik zasiada przed monitorem ekranowym i prowadzi bezpośredni dialog z maszyną.

Zmieniły się komputery, ale też zmienili się i użytkownicy – powoduje to, że również programy muszą spełniać inne wymagania. Przeciętny użytkownik nie jest już informatykiem, lecz fachowcem o zupełnie innej specjalności. Trudno się więc dziwić, że nie chce on zaczynać pracy z nowym narzędziem od głębokich studiów nad oprogramowaniem. Doskonały sprzęt może stać bezproduktywnie, gdy brakuje wygodnego w użyciu oprogramowania. Stąd też ergonomia oprogramowania urosła do rangi podstawowego problemu.

Aby dialog z komputerem mogli prowadzić ludzie nie przeszkoleni w zakresie elektronicznego przetwarzania danych, producenci dostarczają tzw. *oprogramowanie do komunikacji z użytkownikiem*. Pod tym pojęciem należy rozumieć część programu obsługującą monitor ekranowy i klawiaturę. Musi ona gwarantować łatwą obsługę, wspomagać użytkownika wyświetlając uwagi z objaśnieniami, zabezpieczać przed konsekwencjami błędów i dodatkowo jeszcze zapewniać krótki czas odpowiedzi. Wszystkie te wymagania należy uwzględnić już na etapie projektowania programów.

Programiści, od dawna obeznani z komputerem, z trudem potrafią wczuć się w sytuację przeciętnego użytkownika i rzadko uwzględniają istotne czynniki psychologiczne. Przystosowane do istoty ludzkiej, a więc ergonomiczne oprogramowanie zmniejsza napięcie psychiczne, zwiększa wydajność i sprawia, że człowiek łatwo przyzwyczaja się do nowego rodzaju pracy. Dialog z komputerem musi odbywać się w taki sposób, by przypominał zwyczajną rozmowę – zgodnie z nawykami i reakcjami człowieka.



Niestety, daleko jeszcze do osiągnięcia tego celu. Wiele systemów nie zapewnia jednolitej obsługi, nawet w obrębie jednego programu. Prośbę o pomoc (ang. help) – przypomnienie możliwych rozkazów – czasem zgłasza się wciskając klawisz ze znakiem zapytania, a innym razem z literą H. Zależy to od... realizowanego właśnie podprogramu.

Brak jest, jak dotąd, standardów definiujących sposób wydawania najpopularniejszych nawet poleceń (np. korygowania błędów, wstawiania i usuwania znaków itp.). Zbliżone funkcją programy często diametralnie różnią się sposobem obsługi – tak że tylko fachowiec wysokiej klasy ma szansę poradzić sobie z nowym systemem przy pierwszym podejściu. W rezultacie zapoznanie się z nie stosowanym wcześniej oprogramowaniem pochłania sporo czasu, użytkownik popełnia wiele błędów i nic dziwnego, że wielu boi się współpracownika-komputera.

O ile zalecenia ergonomii dotyczące sprzętu można względnie łatwo określić, to znacznie trudniej sprecyzować je dla oprogramowania. Zapewne użytkownicy długo jeszcze będą czekać na normalizację w tej dziedzinie. Niemniej, można już podać kilka zasad, które konsekwentnie stosowane na etapie projektowania programu, znacznie ułatwią życie obsłudze:

- jednorodna prezentacja wszystkich rodzajów danych zwiększa czytelność tekstów
- przejrzystość można również poprawić wprowadzając kontrasty, znaki o różnej wielkości i grubości lub stosując kolory – jednakowe dla podobnych elementów
- identyczne funkcje powinny być wywoływane przez te same rozkazy; najlepiej do tego celu wykorzystywać klawisze funkcyjne.

Niekonsekwentne używanie skrótów może sprawić, że praca z komputerem zamieni się w loterię. Na przykład: w pewnych sytuacjach system oczekuje potwierdzenie zlecenia angielskim słowem *YES* (*tak*), w innych wystarczy sama litera *Y*, jeszcze gdzie indziej trzeba napisać *1*. Tego po prostu nie sposób zapamiętać!

Wyświetlane na ekranie pomocnicze zestawienie możliwych komend (tzw. *menu*) oraz pytania zawierające prawidłowe formy odpowiedzi, zmniejszają objętość podręczników i prawdopodobieństwo popełnienia błędu. W *menu* należy nie tylko podać sposób wywołania, ale też w skrótowej formie przypomnieć, jakie funkcje wymienione komendy realizują. Funkcja **HELP**, powodując wyświetlenie *menu* na ekranie monitora, powinna być dostępna praktycznie w każdym momencie, gdy od użytkownika wymagana jest decyzja.

Niezależnie od tego, system powinien wykazywać znaczną *odporność* na błędy. Przykładowo, zbyt dużą liczbę wprowadzonych spacji (odstępów) program może ignorować. Błędy, których system nie potrafi usunąć, powinny być natychmiast zgłaszane użytkownikowi. Omyłkowe zniszczenie danych musi być w wystarczającym stopniu utrudnione – przed wykonaniem zlecenia usunięcia zbioru z danymi, system powinien zażądać potwierdzenia rozkazu.

Można uniknąć *wkuwania słówek*, jeżeli klawisze funkcyjne zostaną jednoznacznie i przejrzyście opisane. Komunikaty o wykrytych błędach powinny zawierać, zamiast tajemniczych kodów, opis zrozumiały dla przeciętnego użytkownika.

Należy też wykluczyć sytuacje podobne do tej, którą zaraz opiszemy. Przez pół roku pracy z mikrokomputerem urządzenie na ogół sygnalizowało błąd operacyjny numer 58. W załączonej przez producentów tabeli opisującej 104 rodzaje błędów – numer 58 oznaczał: „inne błędy".

micro

JEDEN DLA WSZYSTKICH

Nowoczesne mikrokomputery dysponują szybkościami przetwarzania nie odbiegającymi od możliwości dużych komputerów z lat sześćdziesiątych. Przy pojedynczym użytkowniku taki komputer nie jest więc w pełni obciążony. Dla lepszego wykorzystania mocy obliczeniowej można spróbować wykonywać równolegle kilka zadań. Jeżeli jeden komputer równocześnie pracuje dla wielu użytkowników, jego program sterujący (czyli tzw. *system operacyjny*) powinien realizować wiele funkcji zarządzających i koordynujących. Poniżej analizujemy specyficzne operacje w tego typu systemach zwanych *wielodostępnymi*, określamy wymagania w stosunku do systemu operacyjnego i zestawiamy je z właściwościami obecnie dostępnych rozwiązań.

Od czasu, gdy szybkość realizacji programu przez komputer wielokrotnie przekroczyła szybkość działania urządzeń peryferyjnych (łącznie z klawiaturą), pojawiły się metody umożliwiające wykorzystanie tego nadmiaru mocy przez jego podział na wiele zadań wykonywanych równolegle. Na początku była to metoda pracy *wieloprogramowej*. Pozwalała ona rozdzielić moc obliczeniową przez wydzielanie odcinków czasu dla programów wykonywanych równolegle. Czas oczekiwania komputera na powolne urządzenie peryferyjne lub wprowadzanie danych z klawiatury, był wykorzystywany na równoległe wykonywanie innego programu.

Odmianą tej metody jest tzw. *podział czasu* (ang. time sharing), w którym wielu użytkowników dzieli się mocą obliczeniową komputera. W zasadzie przypomina to sytuację, gdy arcymistrz rozgrywa partię szachów jednocześnie na dwudziestu stolikach swych przeciwników.

Istotne jest to, że w przypadku takiego systemu wieloprogramowego, dzieli się tylko moc obliczeniową komputera, natomiast każdy użytkownik dysponuje własnymi danymi i urządzeniami peryferyjnymi. W turnieju szachowym każdy z uczestników również ma własną szachownicę z figurami i nikt – prócz przeciwnika – nie może zabrać mu nawet pionka. W języku informatycznym oznacza to, że istniejące zasoby sprzętowe systemu komputerowego są przydzielane poszczególnym użytkownikom na prawach wyłączności. Tak więc np. paski listy płac nie mogą być sporządzone na tej samej drukarce, na której właśnie jest drukowany wykaz stanu zapasu materiałów w magazynie. W przeciwnym razie dane inwentaryzacyjne wkrótce pojawiłyby się jako elementy wynagrodzenia!

Istnieją jednak dla odmiany informacje, które są istotne dla wszystkich użytkowników. Tak, jak na lotnisku dla wszystkich pasażerów jest jedna duża tablica świetlna z aktualnymi danymi o lotach, w systemie komputerowym są dane, które mogą być oglądane (odczytywane) przez wielu użytkowników. Żaden użytkownik nie ma jednak prawa dokonywać w nich zmian.

Istnieją też dane, które wpływają z wielu źródeł, a więc mogą być zmieniane. Weźmy jako przykład stan zapasu magazynowego: jeżeli sprzedaż następuje bezpośrednio z magazynu, to każde realizowane zlecenie musi być natychmiast odjęte od aktualnego stanu zapasu. Podobnie, każdy przychód do magazynu należy zaksięgować bezpośrednio po pojawieniu się towaru, aby nie blokować możliwości sprzedaży.

PRZETWARZANIE WSADOWE

W klasycznych systemach komputerowych wszystkie zmiany w zbiorach danych są najpierw rejestrowane i gromadzone, a dopiero potem przetwarzane *metodą wsadową (partiową)*. Osobno księgowane są wszystkie przychody do magazynu, następnie odejmuje się wykaz wydanych towarów i dopiero wtedy powstaje zbiór obrazujący aktualny stan zapasów. Liczne wielkie przedsiębiorstwa przetwarzają nadal swoje dane metodą wsadową. Taki tryb postępowania sprawia, że zapisy księgowe nigdy nie są w pełni aktualne.

Nowoczesne systemy komputerowe pozwalają swym użytkownikom na wykonanie opisanego procesu w *trybie konwersacyjnym*: po otrzymaniu zamówienia sprzedawca poleca komputerowi wypisać na monitorze terminala aktualny stan zapasu magazynowego, ewidencjonuje sprzedaną ilość i poleca wpisać zmieniony stan zapasu ponownie do „karty magazynowej".

Konwersacyjne systemy z dostępem dla wielu użytkowników muszą uwzględniać szczególny rodzaj zarządzania danymi: dają one do dyspozycji swym użytkownikom wspólne zasoby danych i w zależności od rodzaju poszczególnych danych użytkownicy ci mogą je tylko oglądać lub też zmieniać i dopisywać do nich nowe. Program sterujący takiego wielodostępnego komputera – jego *system operacyjny* – musi regulowac i kontrolować równoczesny dostęp wielu użytkowników do jednego zasobu informacji, zwanego *bazą danych*.

Gdy informacje z *bazy danych* przepisywane są równocześnie przez wielu użytkowników, niezbędne są odpowiednie działania koordynujące. Użytkownik nie może bowiem zwracać uwagę na to, że zarejestrował i wpisał do pamięci właśnie *rekord* (zapis dokumentu) numer 15, podczas gdy inny użytkownik opracowuje ten sam rekord numer 15. System operacyjny wielodostępnego komputera konwersacyjnego musi więc umożliwiać wielu użytkownikom równoczesne opracowywanie jednego zbioru danych.

Sytuacja staje się bardziej złożona, jeżeli wielu użytkowników sięga w zbiorze danych równocześnie do tego samego rekordu. W tym przypadku system operacyjny powinien określić kolejność i czas dostępu do zbioru oraz zapewnić, aby rekord opracowywany w danej chwili przez jednego użytkownika, był niedostępny (zablokowany) dla pozostałych użytkowników. W przeciwnym razie mogą zaginąć zmiany, które jeden z użytkowników już rozpoczął, lecz jeszcze nie zapisał w zbiorze.

JEŚLI BRAK 1000 GUMOWYCH NIEDŹWIADKÓW

Przykładowo, przy sprzedaży zabawek sytuacja przebiega następująco: sprzedawca Młynarski otrzymuje zlecenie natychmiastowej dostawy 1000 zielonych gumowych niedźwiadków. Wywołuje więc na ekranie swego monitora odpowiedni rekord zapasu materiałowego, odczytuje go i stwierdza, że zlecenie można zrealizować, ponieważ w magazynie znajduje się jeszcze 5000 niedźwiadków. Przyjmuje więc zlecenie, księguje rozchód i wpisuje do zbioru magazynowego rekord z nowym stanem 4000 sztuk.

W tym samym momencie sprzedawca Kowalski również otrzymał zlecenie na zielone niedźwiadki i zapytał o ten sam rekord zapasu magazynowego. Tam jednak zapisane jest ciągle jeszcze 5000 sztuk, ponieważ Młynarski nie skończył operacji zamawiania. Oczywiście Kowalski cieszy się, że swoje zlecenie na 1000 zielonych niedźwiadków może natychmiast zrealizować. Księguje rozchód i poleca wpisać do rekordu liczbę 4000 jako nowy stan zapasu.

Równolegle Młynarski wprowadza nowy stan 4000. Wkrótce w magazynie brakuje 1000 zielonych niedźwiadków, ponieważ system operacyjny nie poinformował Kowalskiego, że na tym samym rekordzie działa Młynarski, który w karcie zapasu zapisał również swoją transakcję.

Z powyższego przykładu wynika, że w przypadku systemu wielodostępnego ze wspólnymi zbiorami danych, nie wystarczy zablokowanie pojedynczego rekordu przed wielokrotnym dostępem, ale również to, że o fakcie blokady należy poinformować wszystkich zainteresowanych. W przeciwnym razie może powstać sytuacja, określana przez informatyków jako *zakleszczenie* (ang. dead lock). Następuje to wtedy, gdy jeden użytkownik pobiera do opracowania pewien rekord zbioru danych, a następnie prosi o rekord innego zbioru, który jednak w tym momencie opracowuje inny użytkownik (a więc jest on zablokowany). Jeżeli teraz drugi użytkownik zamierza właśnie sięgnąć do rekordu, który został już pobrany przez pierwszego, to obaj będą oczekiwali na zwolnienie rekordu przez partnera – bez powodzenia.

Dodatkowe utrudnienie pojawia się, gdy rekordy są ujęte w większe jednostki – *bloki*

danych. Jeżeli system operacyjny nie wstrzyma operacji odczytu całego bloku danych przez jednego użytkownika, wówczas podczas „zwrotnego" zapisu istnieje możliwość wpisania informacji do rekordów, które w tym czasie zostały zmienione przez pozostałych użytkowników.

JEDEN MOŻE, POZOSTALI CZEKAJĄ

Najprostsza metoda zapobiegania wzajemnym konfliktom polega na tym, aby zablokować cały zbiór danych dla pozostałych użytkowników w czasie, gdy jeden z nich go modyfikuje. Takie blokowanie jednego zbioru z bazy danych określane jest jako *blokada zbioru* (ang. file lock). Jest ono stosowane głównie w systemach, które zostały rozszerzone do postaci systemów wielodostępnych, ponieważ w pierwotnej wersji systemy operacyjne nie były przewidziane do tego rodzaju zadań. W systemach z blokadą zbioru użytkownik musi zakomunikować systemowi operacyjnemu zamiar pracy z pewnym *zbiorem danych*. System *otwiera zbiór i tym* samym automatycznie blokuje go dla wszystkich pozostałych użytkowników. Użytkownik może wtedy pracować na wszystkich danych zbioru (a więc je zmieniać), natomiast wszyscy pozostali muszą w tym czasie czekać. Po zakończeniu opracowania następuje zniesienie blokady, czyli *zamknięcie zbioru*. Dopiero teraz następny użytkownik może realizować swoje działania. Otwieranie i zamykanie zbioru wymagają czasu, a więc systemy z *blokadą zbioru* charakteryzują się długimi czasami oczekiwania, zwłaszcza w sytuacji, gdy wielu użytkowników zamierza jednocześnie pracować na tym samym zbiorze.

W wypadku bardziej „inteligentnych" systemów operacyjnych blokowany jest tylko ten rekord zbioru, który użytkownik aktualnie opracowuje. Metoda ta jest nazywana *blokadą rekordu* (ang. record lock). System komputerowy, który ma być przystosowany do efektywnego przetwarzania konwersacyjnego, powinien zawierać taki mechanizm.

Na szczególną uwagę w systemie wielodostępnym zasługuje zabezpieczenie danych przed ich utratą. System z przetwarzaniem *wsadowym* wychodzi ze zdefiniowanego stanu wszystkich danych, które są akualizowane w jednym – każdorazowo powtarzalnym – przebiegu przetwarzania. W systemie, w którym manipulacje na danych wykonywane są na bieżąco, mogą wystąpić stany nieokreślone. Powstają one wtedy, gdy wystąpi awaria systemu w czasie, w którym informacje zmieniane są w wielu miejscach i przez wielu użytkowników. Konwersacyjny system wielodostępny działa i „pada" z przewidzianymi jednak w takiej sytuacji możliwościami powtórzenia przebiegu – żaden system komputerowy nie jest całkowicie odporny na awarie.

Jeśli system operacyjny prowadzi specjalny *zbiór ewidencyjny (dziennik)*, w którym zapisuje się wszystkie zmiany stanu danych, to zapisy te są wykorzystywane do rekonstrukcji informacji. Jeżeli użytkownik musi samodzielnie obsługiwać wspomniane funkcje ewidencjonowania w dzienniku, różnice w czasowej kolejności transakcji mogą prowadzić do całkowicie różnych wyników faktycznego przetwarzania i przebiegu zrekonstruowanego. Przykład z gumowymi niedźwiadkami wskazuje, że przyjęcie lub odrzucenie zlecenia może zależeć od tego, czy przychód magazynowy został zaksięgowany przed czy po zapytaniu o stan zapasu.

PRAWIDŁOWO, A PRZECIEŻ BŁĘDNIE

Dzięki rejestrowaniu danych dotyczących zmian oraz tematycznemu zabezpieczaniu kolejnych stanów bazy danych, można zagwarantować fizyczne bezpieczeństwo oraz prawidłowość danych. Jednym z założeń jest to, że wszystkie użyte nośniki danych, po ich zapisaniu, zostają zabezpieczone przed możliwością dostępu przez system. Polega to zwykle na tym, że dyskietka lub taśma magnetyczna po pozytywnym zakończeniu zapisu zostaje wyjęta z urządzeń napędowych i umieszczona w szafie pancernej.

Nie można jednak sprawdzić, czy tak zabezpieczone dane są poprawne również pod względem logicznym. Wskutek drobnej awarii w czasie tworzenia kopii bezpieczeństwa mogą wystąpić stany nieokreślone, a wynikające z nich błędy zostaną zauważone dopiero po tygodniach lub miesiącach, gdy dane oryginalne nie będą już istniały.

Dlatego w zastosowaniach konwersacyjnych już przy tworzeniu założeń organizacyjnych należy zaplanować kontrolę poprawności logicznej wyników, jako stały element przetwarzania. W przypadku systemów bez funkcji automatycznego zapisu informacji o zmianach oraz rekonstrukcji stanów, fizyczną i logiczną poprawność danych można osiągnąć jedynie przy rezygnacji z aktualności zapisów. Systemy takie wykluczają tym samym prawdziwe zastosowanie konwersacyjne. Wobec tego wspólne informacje w takim systemie mogą być organizacyjnie dostępne jedynie dla operacji czytania.

AKTUALIZACJA KONTRA BEZPIECZEŃSTWO

Przy wprowadzaniu systemu wielodostępnego z działaniem w trybie konwersacyjnym, każdy użytkownik powinien najpierw sprawdzić, czy korzyści wynikające z możliwości prezentacji aktualnych informacji są we właściwej proporcji z nakładami ponoszonymi na zabezpieczanie danych. W wielu przypadkach wystarcza osobne zapamiętanie wszystkich zmienionych danych oraz przeprowadzenie aktualizacji zbioru w odrębnym przebiegu przetwarzania.

System operacyjny przystosowany do pracy konwersacyjnej z wieloma użytkownikami, powinien spełniać co najmniej następujące wymagania:

- powinna istnieć możliwość równoczesnego zapełniania zbioru danych przez wielu użytkowników,
- rekord, który może być zmieniany, powinien być chroniony przed równoczesnym dostępem wielu użytkowników,
- pojedynczy rekord powinien być blokowany również w przypadku opracowywania większej jednostki danych do której należy,
- o uruchomieniu blokady rekordu powinien być zawiadamiany program organizujący dostęp lub użytkownik,
- dane dotyczące zmian powinny być zapisywane przez system operacyjny w prawidłowej (pod względem chronologicznym) kolejności,
- zapamiętane dane dotyczące zmian należy wykorzystywać do rekonstrukcji baz danych.

Komputery osobiste i systemy wielodostępne to w zasadzie dwie całkowicie przeciwstawne filozofie. Komputer osobisty, jak sama nazwa wskazuje, powinien być przeznaczony do wyłącznej dyspozycji jednego użytkownika. Jeżeli więc komputer osobisty ma być jednocześnie dostępny dla wielu *użytkowników, jest to już sprzeczność sama* w sobie. Oprócz tego, systemy operacyjne komputerów osobistych są skonstruowane specjalnie dla wynikających z nazwy zastosowań (jeden komputer = jeden użytkownik).

Jeżeli jeden mikrokomputer ma obsługiwać równocześnie wielu użytkowników, to musi być on także wyposażony w odpowiedni system operacyjny. System taki rozdziela równomiernie moc obliczeniową, obsługuje

Rys. 1. W systemie z podziałem czasu (ang. time-sharing system) komputer centralny przydziela swój czas „po kawałku" dołączonym do niego terminalom. Są to z reguły terminale „nieinteligentne", a ich wydajność zależy od systemu centralnego. Obsługiwane terminale wybierane są jeden po drugim. Dzięki szybkiemu przełączaniu użytkownicy nie zauważają okresów braku dostępu do komputera centralnego.

monitory ekranowe i klawiatury oraz koordynuje zgłoszenia dotyczące wykorzystania drukarki.

Systemy operacyjne dla komputerów osobistych, dodatkowo rozbudowane pod kątem potrzeb eksploatacji wielodostępnej, z reguły obejmują tylko blokadę zbioru. Oznacza to, że zbiór danych pozostaje tak długo zablokowany dla wszystkich pozostałych użytkowników, jak długo jeden z nich na nim pracuje.

SIECI LOKALNE

Najpowszechniejszą wersją systemu wielodostępnego z użyciem komputerów osobistych jest sieć lokalna. W tym przypadku wiele niezależnych komputerów osobistych łączy się ze sobą wspólnym przewodem. Każdy użytkownik ma wtedy swój własny ekran, własną drukarkę oraz własną bazę danych na dyskietce lub sztywnym dysku.

Rys. 2. W wielokomputerowym systemie z komputerem nadrzędnym (ang. master-slave system) jeden z mikrokomputerów – komputer nadrzędny (ang. master) obsługuje urządzenia zewnętrzne: drukarkę, ploter, pamięć masową (stacje dysków wymiennych i niewymiennych). Pozostałe mikrokomputery są jednak urządzeniami inteligentnymi – mają własne procesory i pamięci robocze. Przetwarzanie wykonują mikrokomputery podległe (ang. slave) – programy, potrzebne w danej chwili, są ściągane z komputera nadrzędnego.

Do tego samego przewodu przyłącza się dodatkowe komputery bez ekranu, które sterują wspólnymi danymi i urządzeniami. Nazywane są one *administratorami zbiorów* bazy danych (ang. file server).

W sieci lokalnej każdy komputer osobisty pracuje całkowicie niezależnie. Jeżeli będzie on chciał sięgnąć do wspólnej bazy danych, to wyśle swe zgłoszenia do administratora bazy, który udostępni mu potrzebne dane, blokując je jednocześnie dla wszystkich pozostałych użytkowników. To samo dzieje się z przyłączonymi do administratora drukarkami, przy czym przekazane polecenia drukowania są przez niego przejściowo zapamiętywane, jeżeli drukarka pracuje w tym momencie dla innego użytkownika.

Bardziej zaawansowane systemy sieciowe znoszą rozdział pomiędzy komputerami zgłaszającymi żądania oraz świadczącymi usługi. W rozwiązaniach tych każdy komputer osobisty może w ramach sieci żądać danych z innych komputerów albo też udostępniać im własne dane. Również urządzenia peryferyjne każdego komputera osobistego udostępniane są dla wszystkich użytkowników, przy czym równocześnie pojawiające się żądania system operacyjny włącza do kolejki oczekiwania i sukcesywnie je realizuje.

Porównując tę metodę z tradycyjnymi rozwiązaniami systemów komputerowych można stwierdzić, że połączone w sieć komputery osobiste zachowują się tak samo, jak duże komputery podczas eksploatacji wielodostępnej. Różnica w stosunku do konwersacyjnych systemów wielodostępnych polega jedynie na braku możliwości stworzenia jednoczesnego dostępu oraz manipulowania zasobami danych dla wielu użytkowników. Przyczyna tkwi w obecnych systemach operacyjnych komputerów osobistych: ani CP/M ani też MS-DOS w zasadzie nie były projektowane dla eksploatacji wielodostępnej.

MIKRO JAKO KOMPUTER KONWERSACYJNY

Jeżeli przy wzajemnym wiązaniu komputerów osobistych w sieć ma być realizowane przetwarzanie konwersacyjne na poziomie rekordów, to system operacyjny powinien być przystosowany do eksploatacji wielodostępnej. Wówczas nie tylko poszczególne komputery osobiste mogą mieć wzajemny dostęp do swoich zasobów poprzez sieć, ale również do takiego komputera wyposażonego w wielodostępny system operacyjny można przyłączyć dodatkowe monitory ekranowe, które będą dzieliły się mocą obliczeniową, urządzeniami peryferyjnymi oraz danymi. Dzięki temu komputer nie jest już komputerem osobistym, lecz małym systemem wielodostępnym do zastosowań konwersacyjnych.

Obecnie coraz częściej pojawia się nazwa **UNIX. UNIX** to system operacyjny dla wielodostępnych systemów przetwarzania danych. Ciekawe, że **UNIX** powstał już dwadzieścia lat temu w firmie **BELL LABORATORIES**, a dziś jest adaptowany do kolejnej generacji mikrokomputerów.

micro

SINCLAIR 128 KB

Od pewnego czasu krążą plotki, że Sinclair zamierza udoskonalić ZX Spectrum. Najwyższa pora, jeśli firma chce pozostać na rynku mikrokomputerów domowych. Najpoważniejsi konkurenci już od pewnego czasu oferują udoskonalone modele mikrokomputerów 8-bitowych. Atari wprowadziła model 130 XT podsumowujący doświadczenia z 600 XL i 800 XL. Amstrad w kilka miesięcy po wprowadzeniu CPC 664 zastąpił go wersją CPC 6128 z pamięcią RAM rozszerzoną do pojemności 128 KB. Najnowszy model Amstrad 8256 zawiera już 256 KB RAM. Commodore lansuje C 128, czyli C 64 rozbudowane o dwa dodatkowe procesory i z pamięcią 128 KB RAM.

Proponowany przez firmę Sinclair ZX Spectrum+ jest już obecnie komputerem o przestarzałej konstrukcji i niewspółmiernie wysokiej cenie. Podpiera się jeszcze dużą liczbą programów, które opracowano na ten komputer. Firmy produkujące oprogramowanie zaczynają jednak powoli odwracać się od Spectrum.

Sinclair oficjalnie nie zapowiedział jeszcze nowego modelu (listopad 85), ale niektórzy producenci oprogramowania otrzymali modele ZX Spectrum+ z przystawką rozszerzającą. Nowe Spectrum zachowuje kompatybilność programową z poprzednim modelem dzięki zachowaniu dotychczasowego modułu pamięci ROM obok modułu nowego zawierającego m.in. udoskonalony interpreter języka BASIC. Pamięć RAM poszerzona została do 128 KB, co pozwoli pisać prawie trzykrotnie dłuższe programy. Dzięki temu niektóre już istniejące programy (np. Neverending Story – firmy Ocean), które składały się z kilku segmentów, będzie można wprowadzić w całości do pamięci mikrokomputera.

Rozszerzeniu ulegną również możliwości dźwiękowe nowego modelu, gdyż wbudowany zostanie układ AY 3-8910, sprawdzony w komputerach standardu MSX i w Amstradach. Mają zostać również poprawione parametry obrazu. Dotychczasowe utrudnienia w operowaniu kolorami przestaną dręczyć użytkowników.

Po krachu finansowym firmy Sinclair bankierzy stawiają jednak twarde warunki: żadnych kredytów na nowe konstrukcje, dopóki magazyny hurtowni nie zostaną opróżnione z zalegających ZX Spectrum. Dla podratowania finansów, Sinclair zdecydował się na znaczną obniżkę ceny mikrokomputera QL – do 200 funtów (w chwili wprowadzenia na rynek QL kosztował 400 funtów). Zainteresowanie mikrokomputerem QL powinno też wzrosnąć w związku z wprowadzeniem na rynek stacji dysków 3,5" przeznaczonych dla tego komputera.

Jeśli fala świątecznych zakupów okaże się dla firmy Sinclair korzystna, planowane jest wprowadzenie na rynek trzech nowych produktów. Obok wspomnianego Spectrum 128, planowany jest QL II z pamięcią RAM 256 lub 512 KB i pakietami programowymi firmy PSION w pamięci stałej. Oczekuje się też, że Sinclair wprowadzi zupełnie nowy model tzw. komputera przenośnego (ang. portable) o nazwie Pandora.

AMSTRAD STORY

Fot. 1. Amstrad CPC 664

Najprawdopodobniej w momencie ukazania się w druku pierwszego numeru MIKROKLANU nie wszystkie informacje w nim zawarte będą aktualne. Firma Amstrad atakuje bowiem potencjalnych klientów coraz to nowym rozwiązaniem, powodując u niektórych niemałe stresy.

Jakże inaczej ma zareagować szczęśliwy nabywca najnowocześniejszego przez kilka ostatnich miesięcy modelu, jeżeli dowiaduje się, że za niewiele wyższą cenę może już nabyć maszynę istotnie lepszą, dysponującą znacznie dłuższą listą oprogramowania?

Mija właśnie półtora roku od rynkowego debiutu Amstrada CPC 464, występującego również pod nazwą Schneider. Start był bardzo udany, sukces rynkowy natychmiastowy. Wobec istnienia ostrej konkurencji – wypadek to zaiste nie zwyczajny. A co się tym kryje?

Otóż chyba przede wszystkim to, że AMSTRAD jest konstrukcją wskroś przemyślaną. Nie ma w nim żadnych nowości ani z zakresu technologii, ani architektury mikrokomputerów. Konstruktorzy zwrócili jednak baczną uwagę na błędy popełnione w wielu wcześniejszych opracowaniach mikrokomputerów innych firm. Takie podejście potwierdza się zresztą w kolejnych modelach Amstradów. Przyjrzyjmy się im bliżej.

Zacznijmy od modelu najstarszego, najtańszego i najbardziej w Polsce popularnego: CPC 464. Podobnie jak i następne modele jest on sprzedawany wraz ze specjalizowanym monitorem TV – kolorowym lub monochromatycznym. Jakość ich nie wzbudzałaby żadnych zastrzeżeń, nawet, gdyby cena zestawu była wyższa. Takie połączenie ma swoje zalety, ma też swoje wady. Jakość obrazu i uniezależnienie się od *rodzinnego* telewizora przemawia za, dodatkowy wydatek – przeciw. Dla tych, którzy jednak zechcą korzystać z telewizora, oferowany jest modulator (w systemie PAL) i zasilacz – zasilanie pochodzi bowiem z monitora.

Kolejną wspólną cechą wszystkich modeli jest zintegrowanie pamięci masowej w jednej obudowie z komputerem. W przypadku CPC 464 jest to specjalnie zaadaptowany magnetofon kasetowy. Jest on bardzo niezawodny, co powoduje, że kłopoty z wczytywaniem programów praktycznie nie występują. Szybkość transmisji do i z magnetofonu wynosi 1 lub 2 kBd.

Możliwości graficzne CPC 464 są porównywalne z innymi mikrokomputerami: rozdzielczość 320×200 punktów w wypadku czterech kolorów lub 160×200 w wypadku szesnastu.

Jeżeli jednak rezygnujemy z tęczy przed oczami – a przecież – mikrokomputerów używamy nie tylko do gier – to otrzymamy do dyspozycji grafikę nie spotykaną w innych konstrukcjach tej klasy: 640×200 punktów, w dalszym ciągu mając możliwość definiowania koloru tła i obrazu. Owe 640 punktów oznacza, że w każdej z 25 linii tekstu na ekranie można wyświetlić 80 znaków, a to jest już klasa sprzętu profesjonalnego.

Jak przystało na urządzenie nowoczesne, możliwości dźwiękowe są imponujące: trzy niezależnie programowane kanały z efektem stereo, możliwym do uzyskania za pomocą łatwo dołączanego wzmacniacza.

Klawiatura: profesjonalna z wyodrębnioną częścią numeryczną i sterowaniem kursora. Przy okazji kursora warto wspomnieć o ciekawostce, jaką jest możliwość kopiowania *kursorem kopiującym* (jednocześnie wciśnięty jest również klawisz COPY), dowolnego fragmentu z ekranu – do miejsca wskazanego przez zwykły kursor.

Mózgiem systemu jest procesor Z80A, który w związku z obsługą ekranu obciążony jest tylko w 75%. Przy okazji należy wspomnieć, że pomimo identyczności procesorów, nie istnieje możliwość przenoszenia oprogramowania z ZX Spectrum. Nie jest to jednak nic dziwnego: od kilka lat istnieje na rynku wiele komputerów opartych na mikroprocesorze 6502 lub jego mutacjach (Commodore, Atari, Apple). Chociaż programy można już liczyć w tysiące, do tej pory nie pojawiła się możliwość przenoszenia oprogramowania.

Pojemność pamięci RAM porównywalna jest z innymi mikrokomputerami. Jest jej 64 KB, z czego ponad 16 KB przeznaczone jest na pamięć ekranu i zmienne systemowe. Pamięć ROM jest jednak większa niż typowa (w tej klasie sprzętu): 32 KB, z czego 16 to interpreter języka BASIC, a 16 to system operacyjny. Z posumowania pojemności pamięci wynika, że jak na możliwości Z80 jest jej trochę za dużo. Jest to częściowo prawda, ponieważ pamięć ROM jest nakładkowana. System operacyjny rezyduje w obszarze od 0 do 3FFFH zaś BASIC od C000H do FFFFH, wymieniając się w przestrzeni adresowej z pamięcią ekranu. W tym to obszarze pamięci można umieścić kolejne moduły pamięci ROM, do ponad 200. Odpowiedni mechanizm umożliwiający przełączanie pomiędzy różnymi modułami, zawarty jest w systemie operacyjnym.

Wyprowadzona na zewnątrz szyna systemowa umożliwia dołączenie własnych konstrukcji lub modułów dostępnych na rynku, takich jak łącze szeregowe czy modem. Szczególnie łatwe jest rozszerzenie konfiguracji CPC 464 o stację dyskietek: firma AMSTRAD oferuje dyskietki 3-calowe (o po-

jemności 180 KB na stronę, z zapisem dwustronnym) jak na razie nietypowe na rynku, lecz mające duże szanse na sukces właśnie dzięki Amstradowi. Pierwszą stację dysków elastycznych nabywa się ze sterownikiem, do którego dołączyć można jeszcze jeden mechanizm napędowy.

Przy pewnej dozie wiedzy można oczywiście dołączyć dyskietki 5 1/4 cala.

A skoro mamy już w zestawie dyskietki, to przydałby się dyskowy system operacyjny. Firma oferuje CP/M 2.2. Nie wszystkie jednak programy pracujące pod tym systemem są do wykorzystania. Wynika to z ograniczonej pojemności pamięci RAM, dostępnej dla programów CP/M (43 KB). Oprócz systemu operacyjnego, nabywca otrzymuje bogaty w możliwości graficzne, interpreter języka Logo.

Komunikację ze światem zewnętrznym umożliwia ponadto wbudowane złącze równoległe typu Centronics, użyteczne w szczególności przy korzystaniu z drukarki.

Oprogramowanie: bardzo szybki interpreter rozszerzonej wersji języka BASIC (mimo iż co czwarty cykl zegara przeznaczony jest na obsługę ekranu). Oprogramowanie układowe, (ang. Firmware) jest w maksymalnym stopniu dostępne dla użytkownia poprzez tablice z adresami procedur, przy czym podobnej zasadzie mogą być dodane procedury użytkownika. Ponadto – 200 tytułów najróżniejszych programów, w większości gier, ale również programów użytkowych, takich jak Assembler, kompilatory języków PASCAL i FORTH (a w wersji dyskowej nawet TURBO PASCAL), bazy danych, programy do przetwarzania tekstów, programy kalkulacyjne.

W maju 1985 roku można już było nabyć nowszy model: CPC 664. Do podstawowych różnic należy zaliczyć zmianę zintegrowanego modułu pamięci zewnętrznej z magnetofonu na stację dyskietek 3-calowych, zachowując jednocześnie możliwość pracy z magnetofonem, który stał się urządzeniem zewnętrznym. Kolejna zmiana, to rozszerzona lista instrukcji BASICA o pewną liczbę instrukcji graficznych i innych.

Kompatybilność obu modeli jest zachowana w górę, tzn. prawie wszystko, co zostało napisane dla modelu 464, powinno się wykonać na modelu 664. Stwierdzenie *prawie wszystko* oznacza, że nie będą się wykonywać programy wykorzystujące obszary pamięci przewidziane dla dysku, które w modelu 464 nie są niczym chronione.

Zachowana została zasada wprowadzania rozszerzeń do BASICA, czyli wszystkie *sztuczki* opracowane wcześniej można wykorzystywać.

I ten model nie zapewnia jednak użytkownikowi pełnej pojemności pamięci dla programów działających pod CP/M. Najpopularniejsze programy aplikacyjne, takie jak *dBase II, Wordstar* czy *Visicalc* nie będą działać. Istnieją jednak ich funkcjonalne odpowiedniki. Oto one:

- dBASE II – *Masterfile, Micropen* (istnieje również taśmowa wersja *Masterfile* oraz inne bazy danych)
- Wordstar – *Tasword, Tasword D, Microscript* (także w wersjach taśmowych – *Tasword* to program pozwalający opracować jeden standardowy tekst z różnymi wstawkami, np. standardowa treść listu do wielu adresatów)
- Visicalc – *Mastercalc, Microspread (Mastercalc* również z wersji taśmowej, co oznacza, że programy można używać na modelu 464).

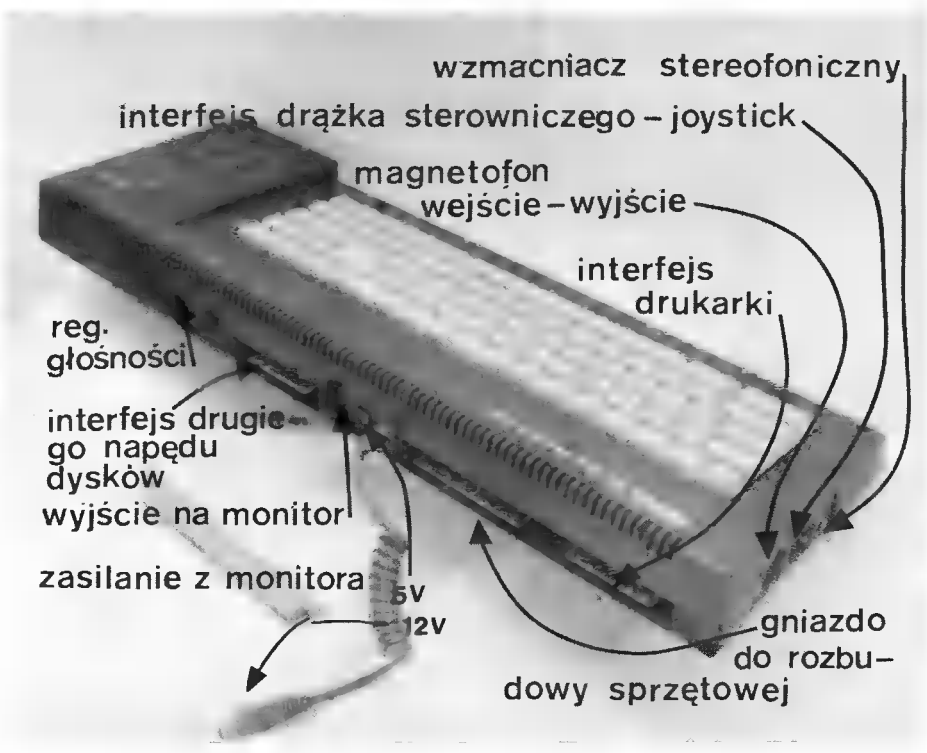
To jednak nie koniec AMSTRAD STORY. Oto w sierpniu ub. roku ukazał się na rynku kolejny model CPC 6128. Najkrótsza charakterystyka tego komputera to: Amstrad CPC 664 + 128 KB RAM. Fakt ten już nie tylko lokalne, firmowe znaczenie. Ustanawia się w ten sposób nowy standard sprzętu mikrokomputerowego. Najprawdopodobniej od tej pory żaden z producentów nie odważy się wejść na rynek z rozwiązaniem, które nie zapewnia użytkownikowi korzystania z dysku i pełnej (tj. 64 KB) pamięci, a w praktyce co najmniej 128 KB, ponieważ jeszcze jest pamięć ekranu i pewne obszary pamięci przeznaczone dla system operacyjny. Można powiedzieć, że wykonany został kolejny duży krok w kierunku większych możliwości i komfortu użytkownika mikrokomputera. Stało się to za sprawą technologii, a mianowicie dzięki bardzo znacznemu spadkowi cen układów pamięciowych (za układy potrzebne do zbudowania 64 KB pamięci płaci się obecnie ok. 10 dol. USA). A zwiększona pamięć operacyjna, to marzenie większości użytkowników, to znacznie szybciej działający komputer.

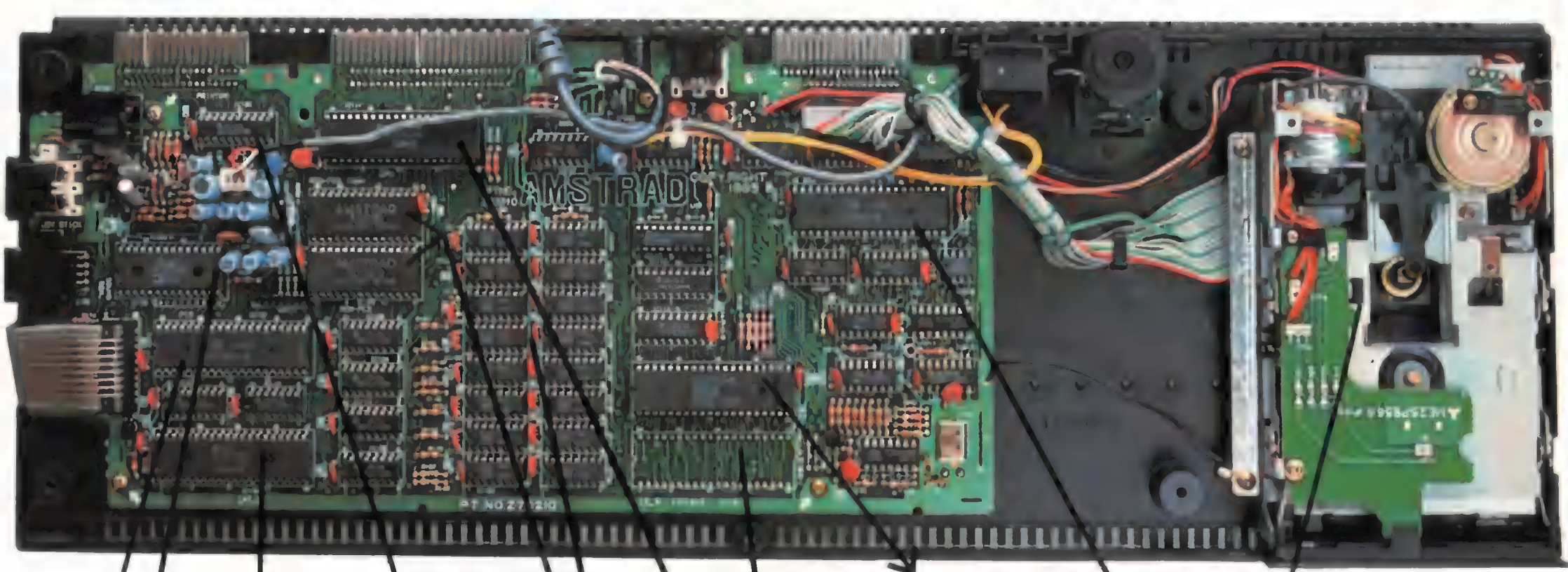
Mając Amstrada 6128 użytkownik może *szaleć* na 128 KB pamięci RAM, wobec czego praktycznie nie istnieją ograniczenia na użytkowanie wszelakiego CP/M-owego oprogramowania – trzeba go tylko mieć. Kompatybilność z oprogramowaniem modeli wcześniejszych jest zachowana: dla programów w BASICU pozostaje tyle samo pamięci, tzn. korzystać można tylko z pierwszych 64 KB pamięci. Ale CP/M, tym razem już w wersji CP/M PLUS (bardziej znanej jako CP/M 3.0), potrzebuje pełnej pamięci. Jest to zresztą wersja pozwalająca na wykorzystanie więcej niż 64 KB pamięci RAM.

Ażeby przyszły użytkownik nie miał wątpliwości, czy warto kupić nowszy model, dostaje jako wyposażenie dodatkowe (ponad to, co oferowane było dotychczas) system CP/M 3.0 oraz program GSX (Graphics System eXtension), przeznaczony do tworzenia grafiki na ekranie, drukarce i ploterze.

Z ciekawostek wspomnieć należy o programie *BANKMANAGER*. Umożliwia on wykorzystanie następnych 64 KB pamięci jako pamięci czterech ekranów (na jeden ekran przypada 16 KB) lub jako zbioru danych. W pierwszym przypadku posługując się instrukcjami *screenswap* i *screencopy* można otrzymywać bardzo szybkie zmiany zawar-

Fot. 2. Amstred 6128 – możliwości współpracy z otoczeniem





generator dźwięku

ULA matryca logiczna

pamięć stała ROM z oprogramowaniem firmowym

napęd dysków

mikroprocesor Z80A

port równoległy 8255 do obsługi klawiatury i układu generacji dźwięku

układ sterownika dysków elastycznych

blok pamięci dynamicznej RAM

sterownik CRT6845

miejsce na inny typ układu ULA

port drukarki

Fot. 5. Amstrad CPC 6128 z otwartą obudową

tości całego ekranu, co jest możliwe dzięki zastosowanemu układowi sterowania ekranu. Z kolei, wykorzystanie instrukcji *bankopen*, *bankread*, *bankwrite* oraz *bankfind* tworzy namiastkę użycia dysku zbudowanego na układach pamięci RAM.

Fot. 3. Amstrad CPC 464

Fot. 4. Amstrad 6128

I wreszcie model, który nie był jeszcze publicznie przedstawiony: JOYCE PCW 8256.

Tym razem nie jest to prosta kontynuacja dotychczasowej linii. Z technicznego punktu widzenia nowością jest kolejne rozszerzenie pamięci o 128 KB, z czego około połowy będzie *udawać* dysk. W efekcie da to znaczne (rzędu kilkudziesięciu razy) przyspieszenie dostępu do danych i programów zapisanych na dyskach. Z zapowiedzi firmy wynika (a jak do tej pory wszystkie swoje zapowiedzi ona urzeczywistniała), że wbudowana stacja dyskietek będzie już z zapisem dwustronnym, tzn. o pojemności 360 KB (po 180 KB na stronę), natomiast druga – stanowiąca rozszerzenie – miec będzie pojemność ok. 720 KB. Niemalże rewolucję zobyczymy na ekranie: 32 wiersze po 90 znaków, czyli o ok. 50% informacji więcej. Klawiatura, oprócz standardowego zestawu, zawierać będzie kilka dodatkowych klawiszy przystosowanych do przetwarzania tekstów, bo model JOYCE został zaprojektowany przede wszystkim do tego celu.

Ma się rozumieć, że program do przetwarzania tekstów wchodzi w skład standardowego wyposażenia, podobnie jak CP/M 3.0, GSX, Logo oraz Mallard extended BASIC (z podwyższoną dokładnością obliczeń i swobodnym dostępem do danych).

Przy tej okazji godzi się wspomnieć, że spośród wielorakich zastosowań mikrokomputerów właśnie przetwarzanie tekstów wiedzie prym, przypada bowiem na nie ok. 40% wszystkich aplikacji. Idąc dalej, najpopularniejszym programem do przetwarzania tekstów jest Wordstar, który w ocenie wartości tego typu programów znajduje się dopiero powyżej czterdziestego miejsca.

W podsumowaniu jeszcze kilka uwag ogólnych: jak do tej pory eksploatowane w kraju AMSTRADY wykazują bardzo wysoką niezawodność, co pokrywa się np. z opinią sprzedawców brytyjskich (najprawdopodobniej ustalona na podstawie liczby komputerów zwróconych w okresie gwarancyjnym). Jedną z wad tego komputera na rynku polskim jest nietypowy format dyskietek, co w znacznym stopniu utrudnia kopiowanie programów z dość obszernej już biblioteki CP/M. I już na sam koniec o pewnej własności Amstrada, która zdecydowanie może zwiększyć zainteresowanie nim: otóż po przyłączeniu stacji dyskietek 5 1/4 cala można go wykorzystywać jako stanowisko wprowadzania danych do profesjonalnego sprzętu klasy IBM PC. Ale to już temat na osobne spotkanie.

MAREK GÓRECKI

PROGRAMATOR PAMIĘCI EPROM

Programator pamięci EPROM jest niemal niezbędnym wyposażeniem każdego systemu mikroprocesorowego. Programator przedstawiony w niniejszym artykule charakteryzuje się bardzo prostą realizacją sprzętową.

Podstawowym elementem programatora jest układ 8255 (MCY 7855). Linie portu B układu doprowadzone są do linii danych pamięci EPROM. Port *A* umożliwia doprowadzenie do układu pamięci linii adresowych *A0* do *A7*. Sygnały na bardziej znaczących liniach adresowych generowane są przez dodatkowy układ licznika 4520 (MCY 74520). Sygnał *PC6* układu 8255 zapewnia zerowanie licznika (aktywny w stanie wysokim), zaś opadające zbocze sygnału *PC7* powoduje zwiększenie zawartości licznika o jeden. Programowe sterowanie tymi dwoma sygnałami umożliwia łatwe ustawienie adresów *A8* do *A12* na dowolną wartość. Dzięki takiej realizacji zaoszczędzono cztery linie adresowe, realizując sterowanie całego programatora jednego tylko układu wejścia-wyjścia. Sygnały *PC2* do *PC5* służą do załączania napięcia na linię Vpp[1]. Wartość tego napięcia zależy od stanu linii *PC* sposób przedstawiony w tabeli.

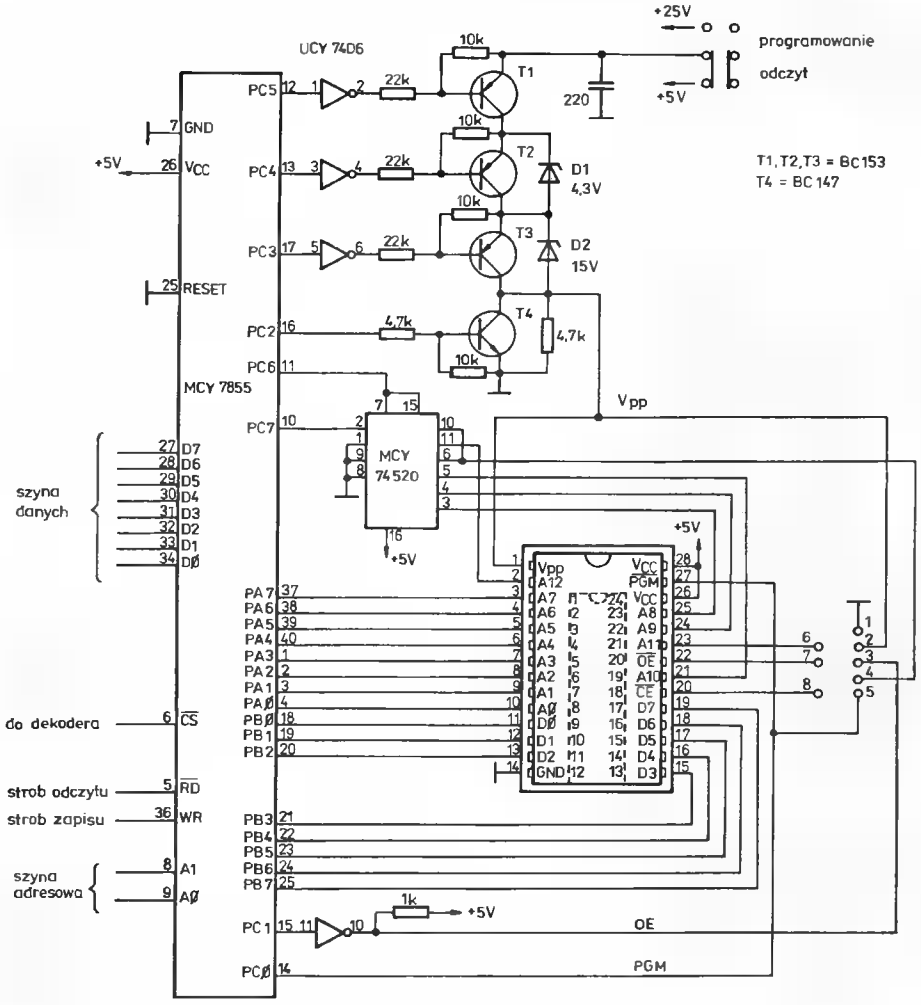
Uruchomienie programatora należy rozpocząć od sprawdzenia poprawności działania układu załączającego napięcie Vpp. W tym celu należy, pomocą prostego programu, wpisać do portu *C* kolejne wartości *PC5...PC2* z tabeli. Należy pamiętać o wcześniejszym zaprogramowaniu portu *C* jako portu wyjściowego. Poprawność połączeń na liniach adresowych i danych, dochodzących do podstawki programatora, można sprawdzić wykorzystując podprogram *EPROM*. W tym celu ustawiając wybrane wartości w rejestrach *BC* i *M(HL)* należy wywołać podprogram *EPROM* i sprawdzić poprawność sygnałów na podstawce pamięci.

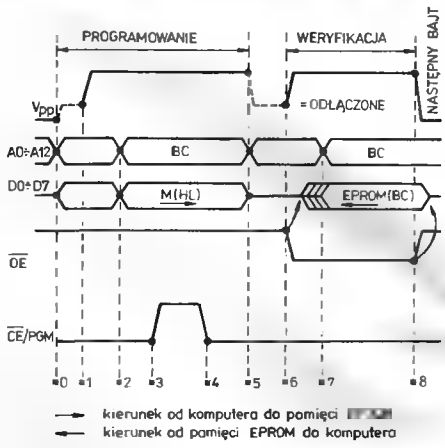
Kolejny ważny krok, to sprawdzenie procedur programujących. Wykonując je krok po kroku należy sprawdzić zgodność każdej procedury z odpowiednimi wykresami czasowymi.

Po wykonaniu powyższych czynności należy napisać program komunikacji z użytkownikiem oraz procedury *DELAY* (odmierzania czasu 45 do 55 ms) oraz *HILO* (wyznaczającej kolejny adres komórki pamięci).

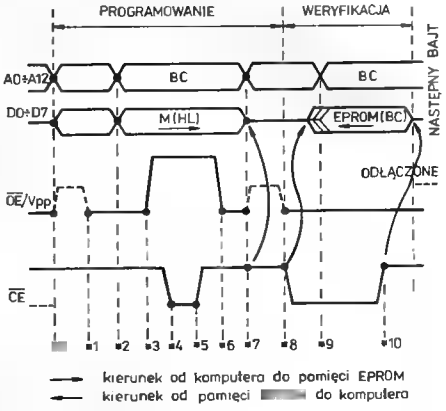
[1]) Wolf P.: EPROM-Programmiergerät. MC Die Mikrocomputer-Zeistschrift 7, 1983

Rys. 1. Schemat ideowy programatora pamięci EPROM

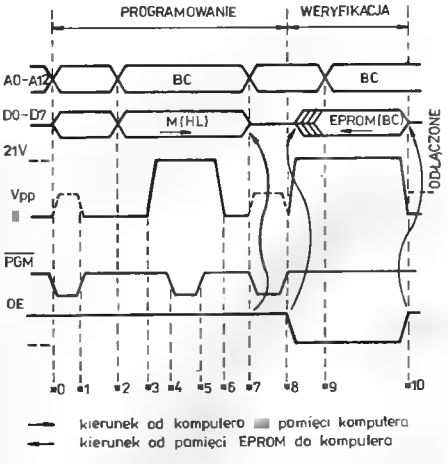




Rys. 2. Wykres czasowy procedury PR2716 – programowanie jednego bajtu



Rys. 3. Wykres czasowy procedury – programowanie jednego bajtu



Rys. 4. Wykres czasowy procedury PR2764 – programowanie jednego bajtu

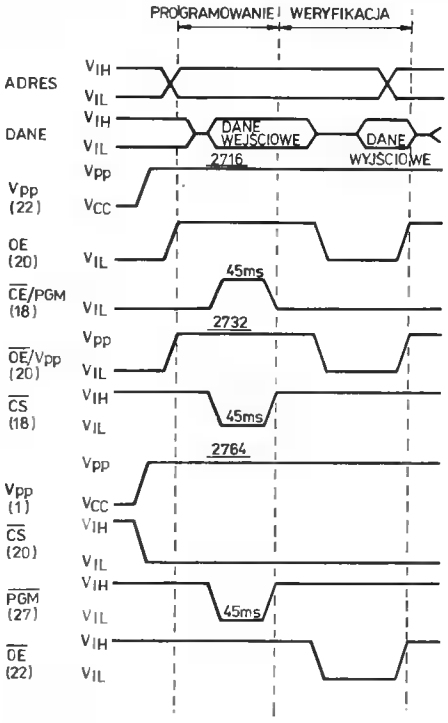
Program powinien rozpoczynać się od ustawienia wskaźnika stosu, zaprogramowania układu 8255 (port *A* i *C* jako wyjście oraz port jako wejście) oraz wyświetlenia zgłoszenia.

Wszystkie przedstawione procedury programujące sprawdzają poprawnosc programowania bajt po bajcie, ustawiając wskaźnik zera *Z=1*, gdy programowanie odbywa się w sposób prawidłowy, oraz *Z=0* w przypadku wystąpienia błędu.

Nie trzeba chyba nikogo przekonywać, że oprócz procedur programujących pożądane

```
'PROCEDURY PROGRAMOWANIA EPROMOW'       PAGE    1

                        TITLE 'PROCEDURY PROGRAMOWANIA EPROMOW'
                        PUBLIC  DELAY,HILO
                        PUBLIC  CONTR,PC,PB,PA
                        ;*****************************************
                        ;* PROCEDURA PROGRAMUJACA DLA 2716      *
                        ;*****************************************
                        ; POLACZONE STYKI 2-6,3-7,5-8
                        ; WEJ HL - ADRES POCZATKU BLOKU RAM
                        ;      DE - ADRES KONCA BLOKU RAM
                        ;      BC - ADRES POCZATKU PAMIECI EPROM
                        ; WYJ A=0 - BLAD
                        ;      A=DANA - DANA DO ZAPROGRAMOWANIA
                        ;      A<>  - DANA ODCZYTANA Z PAMIECI EPROM
                        ;      A=FF - POPRAWNE ZAKONCZENIE
                        ;
                        ; ZALACZANIE/WYLACZANIE SYGNALOW PROGRAMATORA:
0038                    VPPS    SET     00111000B       ; VPP=25V
0004                    VPPR    SET     00000100B       ; VPP=0V
0008                    OES     SET     00000000B       ; OE='HIGH'
0002                    OER     SET     00000010B       ; OE='LOW'
0001                    PGMS    SET     00000001B       ; PGM='HIGH'
0000                    PGMR    SET     00000000B       ; PGM='LOW'
                        ; SLOWA STERUJACE UKLADEM MCY 7855
0080                    PROGR   SET     080H    ; PA,PB,PC - WYJSCIA
0082                    VERYF   SET     082H    ; PA,PC - WY.,PB - WE.
                        ;
0000' 3E 80     PR2716: MVI     A,PROGR ; $0
0002' D3 0B             OUT     CONTR   ; PA,PB,PC - WYJBCIA
0004' 3E 38             MVI     A,VPPS+OES+PGMR ; $1
0006' D3 00             OUT     PC      ; VPP=25V,OE='LOW'
0008' CD 00B6'          CALL    EPROM   ; $2 - USTAWIENIE ADRESU
000B' 00                XTHL            ; OPOZNIENIE OK.20MIKROSEK.
000C' E3                XTHL
000D' 3E 39             MVI     A,VPPS+OES+PGMS ; $3
000F' D3 00             OUT     PC      ; PGM='HIGH'
0011' CD 0000           CALL    DELAY   ; OPOZNIENIE 50MS
0014' 3E 38             MVI     A,VPPS+OES+PGMR ; $4
0016' D3 0B             OUT     PC      ; PGM='LOW'
0018' 00                XTHL
0019' E3                XTHL            ; OPOZNIENIE
                        ; WERYFIKACJA ZAPISANEGO BAJTU
001A' 3E 82             MVI     A,VERYF ; $5
001C' D3 0B             OUT     CONTR   ; PB - WEJBCIE
001E' 3E 3A             MVI     A,VPPS+OER+PGMR ; $6

0020' 00 00             OUT     PC      ; OE='LOW'
0022' CD 00B6'          CALL    EPROM   ; $7 - USTAWIENIE ADRESU
0025' DB 00             IN      PB      ; $8
0027' BE                CMP     M       ; SPRAWDZENIE POPRAWNOSCI
0028' F5                PUSH    PSW     ; OCHRONA REJ.A I PSW
0029' 3E 04             MVI     A,VPPR+OES+PGMR ; $9
002B' D3 00             OUT     PC      ; VPP=0V
002D' F1                POP     PSW
002E' C2                JNZ             ; Z=0 - WYJBCIE Z BLEDEM
                        ; WYLICZANIE KOLEJNEGO ADRESU
002F' CD 0000           CALL    HILO    ; HL+1
0032' D2 0000'          JNC     PR2716  ; CY=0 GDY DE >LUB= HL
0035' AF                XRA     A       ; Z=1
0036' C9                RET             ; ZAKONCZENIE PROGRAMOWANIA
                        ;*****************************************
                        ;* PROCEDURA PROGRAMUJACA DLA 2732      *
                        ;*****************************************
                        ; POLACZONE STYKI 2-7,3-8,4-6
                        ; PODANIE STANU WE/WY JAK DLA 2716
                        ; ZMIANY SYGNALOW STALYCH:
0038                    VPPS    SET     00111000B ; VPP=25V DLA 2732
0028                    VPPS    SET     00101000B ; VPP=21V DLA 2732A
0000                    OES     SET     00000000B ; CE='HIGH'
0001                    CER     SET     00000001B ; CE='LOW'
                        ;
0037' 3E 80     PR2732: MVI     A,PROGR ; $0
0039' D3 0B             OUT     CONTR   ; PA,PB,PC - WYJSCIA
003B' 3E 04             MVI     A,VPPR+CES      ; $1
003D' D3 00             OUT     PC      ; VPP=0V CE='HIGH'
003F' CD 00B6'          CALL    EPROM   ; $2 USTAWIENIE ADRESU
0042' 3E 28             MVI     A,VPPS+CES      ; $3
0044' D3 00             OUT     PC      ; VPP=21V(LUB 25V)
0046' E3                XTHL            ; OPOZNIENIE
0047' E3                XTHL
0048' 3E 29             MVI     A,VPPS+CER      ; $4
004A' D3 00             OUT     PC      ; CE='LOW'
004C' CD 0000           CALL    DELAY   ; OPOZNIENIE 50 MILISEC.
004F' 3E 28             MVI     A,VPPS+CES      ; $5
0051' D3 0B             OUT     PC      ; CE='HIGH'
0053' E3                XTHL
0054' E3                XTHL            ; OPOZNIENIE
0055' 3E 04             MVI     A,VPPR+CEB      ; $6
0057' D3 00             OUT     PC      ; VPP=0V
005A' E3                XTHL
005A' E3                XTHL            ; OPOZNIENIE
                        ; WERYFIKACJA ZAPISANEGO BAJTU
005B' 3E 82             MVI     A,VERYF ; $7
```

```
'PROCEDURY PROGRAMOWANIA EPROMOW'       PAGE    2

005D' D3 0B             OUT     CONTR   ; PB - WEJBCIE
005F' 3E 05             MVI     A,VPPR+CER      ; $8
0061' D3 00             OUT     PC      ; CE='LOW'
0063' CD 00B6'          CALL    EPROM   ; $9 USTAWIENIE ADRESU
0066' 0B 0B             IN      PB
0068' BE                CMP     M       ; SPRAWDZENIE POPRAWNOSCI
0069' F5                PUSH    PSW     ; ZACHOWAJ REJ.A I PSW
006A' 3E 04             MVI     A,VPPR+CEB
006C' D3 00             OUT     PC      ; $10 CE='HIGH'
006E' C2                RNZ             ; Z=0 - WYJSCIE Z BLEDEM
                        ; PRZYGOTOWANIE KOLEJNEGO BAJTU
006F' CD 0000           CALL    HILO    ; HL+1
0072' D2 0037'          JNC     PR2732  ; CY=0 GDY DE>LUB=HL
0075' AF                XRA     A       ; Z=1
0076' C9                RET             ; POWROT BEZBLEDNY
                        ;*****************************************
                        ;* PROCEDURA PROGRAMUJACA DLA 2716      *
                        ;*****************************************
                        ; POLACZONE STYKI 1-8,3-7,4-6
                        ; WEJ/WY ZAN DLA UKLADU 2716
                        ; ZMIANY SYGNALOW STERUJACYCH:
0028                    VPPS    SET     00101000B ; VPP=21V
                        ;
0077' 3E 80     PR2716: MVI     A,PROGR ; $0
0079' D3 0B             OUT     CONTR   ; PA,PB,PC - WYJSCIA
007B' 3E 0B             MVI     A,VPPR+PGMS+OES ; $1
007D' 0B 00             OUT     PC      ; VPP=0,PGM,OE='HIGH'
007F' CD 00B6'          CALL    EPROM   ; $2 USTAWIENIE ADRESU
0082' 3E 29             MVI     A,VPPS+PGMS+OES ; $3
0084' D3 00             OUT     PC      ; VPP=21V
0086' E3                XTHL
0087' E3                XTHL            ; OPOZNIENIE
0088' 3E 28             MVI     A,VPPS+PGMR+OES ; $4
008A' D3 00             OUT     PC      ; PGM='LOW'
008C' CD 0000           CALL    DELAY   ; OPOZNIENIE 50 MILISEK.
008F' 3E 2F             MVI     A,VPPS+PGMS+OES ; $5
0091' D3 00             OUT     PC      ; PGM='HIGH'
0093' E3                XTHL
0094' E3                XTHL            ; OPOZNIENIE
0095' 3E 04             MVI     A,VPPR+PGMS+OES ; $6
0097' D3 00             OUT     PC      ; VPP=0
                        ; WERYFIKACJA ZAPISANEGO BAJTU
0099' 3E 82             MVI     A,VERYF ; $7
009B' D3 00             OUT     CONTR   ; PB - WEJBCIE

009D' 3E 2B             MVI     A,VPPS+PGMS+OER ; $8
009F' D3 00             OUT     PC      ; VPP=21V,OE='LOW',PGM='HIGH'
00A1' 00 00B6'          CALL    EPROM   ; $9 USTAWIENIE ADRESU
00A4' 0B 00             IN      PB
00A6' BE                CMP     M       ; SPRAWDZENIE POPRAWNOSCI
00A7' F5                PUSH    PSW     ; OCHRONA REJ.A I PSW
00A8' 3E 0B             MVI     A,VPPR+PGMS+OES ; $10
00AA' D3 00             OUT     PC      ; VPP=0,PGM,OE='HIGH'
00AC' F1                POP     PSW
00AD' C9                RNZ             ; Z=0 - WYJBCIE Z BLEDEM
                        ; PRZYGOTOWANIE KOLEJNEGO BAJTU
00AE' CD 0000           CALL    HILO    ; HL+1
00B1' 00 0077           JNC     PR2716  ; CY=0 GDY DE>LUB=HL
00B4' AF                XRA     A
00B5' C9                RET             ; ZAKONCZENIE BEZBLEDNE
                        ;*****************************************
                        ;* PODPROGRAM USTAW. ADRESU I DANYCH    *
                        ;*****************************************
                        ; WEJ BC - ADRES KOMORKI EPROM
                        ;      M(HL) - DANA DO ZAPROGRAMOWANIA
                        ; STALE:
00B6                    SETB6   EQU     00001101B ; PC6=1
000C                    RESB6   EQU     00001100B ; PC6=0
000F                    SETB7   EQU     00001111B ; PC7=1
000E                    RESB7   EQU     00001110B ; PC7=0
                        ;
00B6' 3E 0D     EPROM:  MVI     A,SETB6
00B8' D3 0B             OUT     CONTR   ; ZEROWANIE ADRESU
00BA' 3E 0C             MVI     A,RESB6
00BC' D3 0B             OUT     CONTR
00BE' 79                MOV     A,C     ; USTAWIENIE ADRESOW A0-A7
00BF' D3 00             OUT     PC
00C1' 78                MOV     A,B
00C2' E6 1F             ANI     00011111B ; MAX 8KB DLA 2764
00C4' 3C                INR     A
00C5' 3D        EPRO1:  DCR     A
00C6' CA 00D4'          JZ      EPRO2   ; KONIEC ?
00C9' 3E 0F             MVI     A,SETB7
00CB' D3 00             OUT     PC      ; ZAPAL I ZGAS PC7
00CD' 3E 0E             MVI     A,RESB7
00CF' D3 0B             OUT     CONTR   ; DLA ZWIEKSZENIA LICZNIKA
00D1' C3 00C5'          JMP     EPRO1
00D4' 7E        EPRO2:  MOV     A,M     ; USTAWIENIE DANEJ
00D5' D3 0B             OUT     PB      ; DO ZAPROGRAMOWANIA
00D7' C9                RET

                        END
```



Rys. 5. Przebiegi czasowe programowania pamięci 2716, 2732 i 2764

PC5	PC4	PC3	PC2	Vpp <V>	Typ programowanego układu pamięci EPROM
0	0	0	0	Odłączone	————
1	1	1	0	24,4	2716, 2732
1	0	1	0	20,5	2732A, 2764
1	0	0	0	5,5	————
0	x	x	1	ok. 0	————
1	x	x	1	Stany niedozwolone przy załączonym napięciu zasilającym + 25 V	

Napięcia programujące w zależności od stanów logicznych na liniach PC2...PC5

byłoby napisanie takich procedur, jak:
– sprawdzanie, czy pamięć jest wykasowana, tzn. czy określone lub wszystkie komórki pamięci mają wartość FFH,
– przesyłanie zawartości pamięci EPROM do pamięci RAM komputera,
– porównywanie zawartości pamięci EPROM z zawartością pamięci RAM komputera.
Powyższe procedury można napisać, przy założeniu, że przełącznik odłączający napięcie jest wciśnięty i podaje na linię Vpp napięcie bezpieczne +5V – przy odpowiednim ustawieniu linii *PC2* do *PC5*.

PRAKTYCZNE ZALECENIA

- Źródło napięcia programującego powinno zawierać układ ograniczenia prądu do wartości 50 mA.
- Pierwsze włożenie pamięci EPROM do podstawki może być dokonane dopiero po wyświetleniu komunikatu zgłoszenia, tj. po ustawieniu portu C jako wyjściowego.
- Pamięć EPROM można wkładać i wyjmować tylko wtedy, gdy nie są wykonywane procedury programowania.
- Nie wolno przerywać wykonywania procedur programujących poprzez zerowanie systemu.
- W czasie analizy przebiegów czasowych należy pamiętać, że każda zmiana konfiguracji układu 8255 powoduje zerowanie portów wyjściowych.
- Użyte w prezentowanych procedurach stałe *PA*, *PB*, *PC* i *CONTR* to adresy odpowiednich portów w układzie MCY 7855.

STANISŁAW GARDYNIK

DZIAŁ SPRZĘTU

W systemie mikroprocesorowym część urządzenia, która określa, jakiego typu programy mogą być przez ten system wykonywane, można umownie nazwać sprzętem (ang. hardware). Odpowiednie połączenie układów cyfrowych o wielkim i średnim stopniu scalenia dość dokładnie definiuje skalę problemów, które mogą być rozwiązane przez określony system mikroprocesorowy. Oczywiście funkcje, jakie system realizuje, zależą bezpośrednio od programu zapamiętanego w pamięci systemu, ale rodzaj programu, jaki można do tej pamięci wprowadzić, zależy od elektrycznych połączeń wykonanych na poziomie sprzętu. Uniwersalność konstrukcji, jaką daje zastosowanie mikroprocesora, nie jest uniwersalnością absolutną, lecz dotyczy pewnych zbliżonych do siebie klas problemów. Nie wydaje się celowe stosowanie rozbudowanego systemu do realizecji prostych funkcji sterowania, jak również byłaby realizacja skomplikowanych obliczeń arytmetycznych w prostym sterowniku mikroprocesorowym. Są to przypadki skrajne, lecz uzasadniają tezę, że system mikroprocesorowy już na poziomie sprzętu powinien być dostosowany do funkcji, jaką ma pełnić. Z tego punktu widzenia można stwierdzić, że nie ma jednego, uniwersalnego mikrokomputera, który mógłby zaspokoić wszystkich użytkowników. Dzięki temu wielu elektroników może tu mieć szerokie pole do działania.

Stosunkowo najprostszą częścią komputera jest fragment obejmujący procesor i pamięć. Zależy on od typu procesora, który z reguły narzuca odpowiednią konfigurację swojego najbliższego otoczenia. Więcej uwagi pragniemy poświęcić innym blokom wchodzącym w skład systemu, obejmującym specjalizowane układy łączące świat zewnętrzny z jednostką centralną. W zasadniczy sposób zależą one od funkcji, jaką mają realizować, a przez to wykazują znaczne zróżnicowanie.

W systemie mikroprocesorowym nie jest jednak możliwe całkowite oddzielenie sprzętu od oprogramowania. Aby układ mógł realizować narzucone przez konstruktora funkcje, konieczne jest, by choć część programu znajdowała się w pamięci systemu, natychmiast po załączeniu zasilania. Najprościej można to zrealizować, umieszczając tę część programu w pamięci stałej typu EPROM. Dlatego na początek prezentujemy układ oraz jądro programu obsługi programatora najbardziej popularnych układów pamięci EPROM.

ADAM PLUTA

Pierwsze wersje systemu operacyjnego Unix powstały w latach 1969-1970 dla komputerów PDP-7 i PDP-9, w Bell Laboratories (Murray Hill, stan New Jersey, USA). Kolejne odmiany napisano dla serii minikomputerów PDP-11, przechodząc równocześnie, na ile to było możliwe, z języka asemblera na język C. Dzięki temu zapewniono w znacznym stopniu przenośność systemu Unix na różne komputery. Obecnie rozpowszechnił się on zarówno w kierunku tzw. superminikomputerów (do VAX-11/780 włącznie) jak i mikrokomputerów, szczególnie 16-bitowych, opartych na mikroprocesorach firm Motorola, Zilog i in. Powstało też wiele odmian systemu Unix, opatrzonych innymi nazwami, z których najważniejszy jest Xenix, przeznaczony dla 16-bitowych mikrokomputerów opartych na mikroprocesorach firmy Intel.

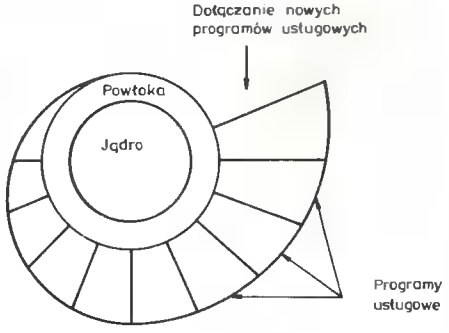
Inną cechą sprzyjającą rozpowszechnieniu systemu Unix było istnienie znacznej liczby programów usługowych, które dziś nazywa się oprogramowaniem narzędziowym, służącym przede wszystkim do wytwarzania oprogramowania i opracowywania różnego rodzaju dokumentacji. Dzięki temu Unix stał się jednym z pierwszych kompletnych środowisk programowych.

Obecnie, bez większego ryzyka można twierdzić, że Unix uzyskał trwałą pozycję na światowym rynku mini- i mikrokomputerowym. Choć niewątpliwie ma wiele wad, sądzi się, że zachowa on tę pozycję co najmniej do czasu upowszechnienia odmiennej architektury komputerów.

Ze względu na coraz większe zainteresowanie tym systemem operacyjnym w Polsce, przy jednoczesnym kompletnym braku literatury polskojęzycznej, w prezentowanym cyklu artykułów podjęta zostanie próba przedstawienia w miarę szczegółowo cech systemu Unix, przy zachowaniu popularnego charakteru opisu.

BUDOWA

System Unix składa się z jądra, tzw. *powłoki* (ang. shell) – będącej w istocie interpreterem poleceń – i dużego zbioru ponad 150 programów usługowych (rys. 1). Do podstawowych funkcji jądra należy:
- zarządzanie procesami
- obsługa wejścia-wyjścia łącznie z zarządzaniem obsługą plików
- wykonywanie operacji zależnych maszynowo.



Rys. 1. Budowa Unixa

CZYM JEST UNIX?

SYSTEM PLIKÓW

Powłoka służy przede wszystkim do zapewnienia właściwej komunikacji z użytkownikiem (operatorem), natomiast programy usługowe stanowią podstawowe narzędzie pracy użytkownika, korzystającego w znacznym stopniu z udogodnień dwóch niższych warstw systemu. Każdy program usługowy jest wywoływany odpowiednim poleceniem *powłoki* (będącym jednocześnie nazwą programu). Ponieważ użytkowanie Unixa wiąże się głównie z wykorzystywaniem programów usługowych, omawiając poszczególne elementy systemu będziemy stopniowo wprowadzać odpowiednie polecenia powłoki.

SYSTEM PLIKÓW

Obiekty, którymi operują programy usługowe, mają swoją reprezentację w tzw. *systemie plików* (ang. file system), stanowiącym jeden z ważniejszych elementów Unixa. Twierdzi się, że utworzenie systemu plików było pierwszym celem przy opracowywaniu Unixa.

Informacja zewnętrzna o plikach (tj. nazwa, wielkość, właściciel, czas powstania i prawa dostępu) jest zapisana w *skorowidzu* (ang. directory), który sam jest plikiem, a informacja wewnętrzna (dane, program, tekst dokumentu itp.) stanowi zawartość pliku. Interesujące jest, że plik w systemie Unix nie ma struktury wewnętrznej, w szczególności nie dzieli się na mniejsze jednostki. Jest uważany za ciąg bajtów, zakończony znakiem EOF (ang. End-Of-File, Control-D, ASCII 004-ósemkowo), niezależnie od tego, jakie jest ich znaczenie. Jedynymi bajtami zawartymi w pliku są bajty umieszczone tam przez użytkownika. Ponieważ nie ma wstępnej alokacji miejsca na dysku, plik może być dowolnej długości, w szczególności może zajmować cały dysk.

Oprócz plików zwykłych (ang. ordinary files) i skorowidzów istnieją pliki specjalne, które stanowią systemową reprezentację urządzeń zewnętrznych. Różnią się one tym od pozostałych plików, że choć ich zawartość jest przechowywana w systemie plików to jednak służą do wymiany informacji z urządzeniami zewnętrznymi, tj. do dostarczania informacji z (lub do) urządzeń zewnętrznych.

ORGANIZACJA HIERARCHICZNA

W systemie Unix system plików jest zorganizowany hierarchicznie, tzn. każdy plik wyższego poziomu może zawierać odsyłacze do plików niższego poziomu, a każdy plik niższego poziomu musi znajdować się w wykazie co najmniej jednego pliku wyższego poziomu, przy czym skorowidz musi znajdować się w wykazie dokładnie jednego pliku wyższego poziomu (każdy węzeł wewnętrzny drzewa jest skorowidzem)

W praktyce, skorowidzem nadrzędnym wobec wszystkich innych jest tzw. *skorowidz pierwotny* (ang. root), zawierający wykaz skorowidzów kilku podstawowych rodzajów:

bin – podprogramy usługowe w wersji gotowej do wykonywania (binarnej),
dev – pliki odpowiadające urządzeniom zewnętrznym,
etc – dane i programy podlegające szczególnej ochronie,
lib – programy biblioteczne, związane z językami wysokiego poziomu, wywołaniami systemowymi, obsługą wejścia-wyjścia itp.
tmp/– pliki tymczasowe tzn. mające ograniczony czas istnienia, używane przez inne programy,
usr – wykaz wszystkich skorowidzów niższego poziomu, w których przechowują swoje pliki użytkownicy, zazwyczaj sporządzony według nazw użytkowników,
inne skorowidze – np. lost+found, zawierający pliki utracone, które straciły właściwe powiązania z systemem plików. Oprócz wymienionych skorowidzów, w skorowidzu pierwotnym znajduje się jeden plik – o nazwie *unix* – nie będący skorowidzem. Jest to jądro systemu operacyjnego tj. program, który może być wykonany przez użytkownika mającego dostęp do skorowidza pierwotnego (ang. root). Takie uprawnienie ma tylko właściciel systemu.

Uprawniony użytkownik może utworzyć sobie w skorowidzu *usr* plik o nazwie *zal*. Jest to skorowidz zawierający wykaz programów i innych skorowidzów tego właśnie użytkownika np. *sm4, intel, multibus,* zawierających także jego programy lub skorowidze z niższych poziomów (rys. 2). W każdym skorowidzu (z wyjątkiem pierwotnego) występuje ponadto element domyślny, stanowiący nazwę skorowidza nadrzędnego (ang. parent), oznaczany dwoma kropkami (..), oraz inny element stanowiący nazwę skorowidza, w którym on sam się znajduje, oznaczony pojedynczą kropką (.).

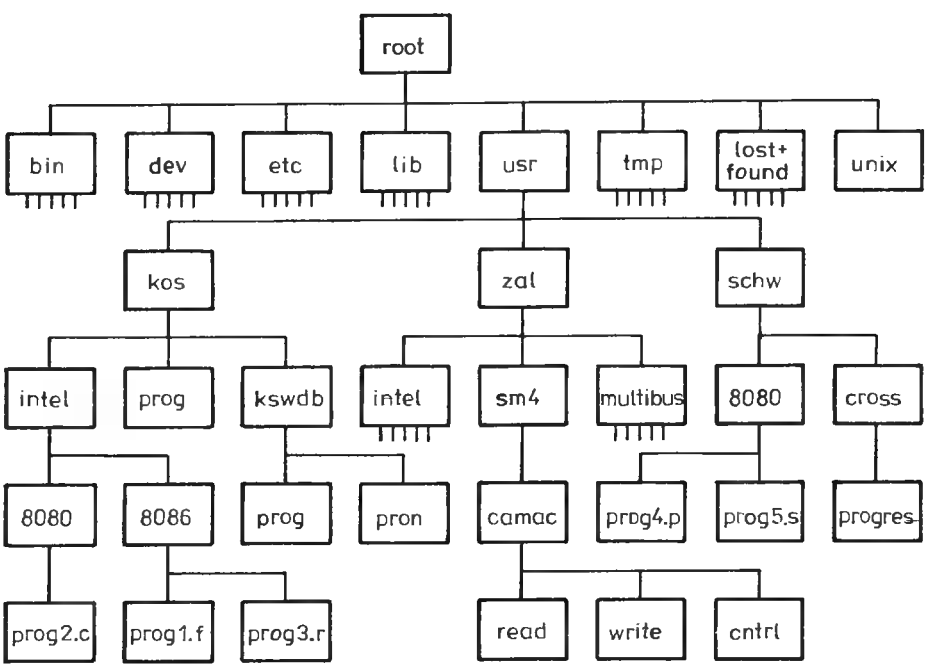
Zasada odwoływania się do plików ułożonych w opisanej strukturze hierarchicznej polega na podaniu pełnej nazwy ścieżki prowadzącej do odnośnego pliku. Przykładowo, ze skorowidza *usr* do pliku *read* prowadzi ścieżka: *usr, zal, sm4, camac, read,* co w notacji systemu Unix zapisuje się jako:
/usr/zal/sm4/camac/read

Po wydaniu polecenia z takim argumentem następuje wykonanie odpowiedniego programu zapisanego w pliku o nazwie *read,* umieszczonym w wykazie skorowidza *camac*. Jeżeli zaś ze skorowidza *camac* chcemy wydać polecenie dostępu do pliku *prog* w skorowidzu *kos,* to powinno ono zawierać argument:
../../../kos/prog
Oczywiście, do wykonania programu *read* bezpośrednio ze skorowidza *camac,* wystarcza użycie polecenia:
read

ZASADY NAZYWANIA PLIKÓW

W tym miejscu warto podać podstawowe informacje o dopuszczalnych nazwach plików. Nazwa pliku może składać się z 14 znaków, teoretycznie dowolnych znaków ASCII (z wyjątkiem znaku pustego, ang. blank). Jednak w praktyce, ze względu na

Rys. 2. Przykładowa struktura systemu plików



specjalne znaczenie niektórych znakow w systemie Unix nie należy używać w nazwie pliku następujących znaków:
/,–,*,?,[,],>,<,|,&,;
Nie należy też zaczynać nazwy kropką (w niektórych realizacjach kropka na początku nazwy oznacza plik ukryty). Praktycznie zaleca się tworzenie nazw plików jedynie z liter, cyfr, znaku podkreślenia i kropki. Należy przy tym pamiętać, że w systemie Unix duże i małe litery są odróżnialne, w przeciwieństwie do większości innych systemów operacyjnych, dlatego dodatkowa reguła zaleca używanie tylko małych liter.

Kropka w nazwie pliku służy do wyróżnienia niektórych rodzajów plików, rozpoznawanych domyślnie przez różne programy usługowe, choć jej stosowanie w tych przypadkach nie jest obowiązkowe. W poniższym wykazie podano typowe znaczenie niektórych końcówek nazw:

.f – programy źródłowe w Fortranie, np. prog1.f

.c – programy źródłowe w języku C, np. prog2.c

.r – programy źródłowe w Ratforze, np. prog3.r

.p – programy źródłowe w Pascalu, np. prog4.p

.s – programy źródłowe w języku asemblera, np. prog5.s

.o programy wynikowe wytwarzane przez translatory z programów źródłowych (np. prog2.o z programu prog.2c)

.h – dane początkowe (nagłówek) dla programu w języku C.

Należy podkreślić, że zakończenie nazwy pliku w ten sposób nie przesądza o typie pliku (ang. extension). Nazwa pliku jest jego pełnym identyfikatorem, a nie jego częścią jak np. w systemie RSX-11, gdzie identyfikator składa się z nazwy, typu i numeru wersji.

Nazwy plików stanowią argumenty wielu poleceń (ang. commands), dlatego aby ułatwić operowanie plikami używa się tzw. *nazw zmiennych* (ang. wildcards) mogących oznaczać więcej niż jeden plik. Możliwe są trzy sposoby tworzenia takich nazw:

- gwiazdka (*) użyta w nazwie zastępuje dowolny ciąg znaków, np. prog* – może oznaczać dowolną nazwę programu spośród następujących: prog, prog5, progi, prognostyk, prognoza, program, programator, progres itp.,
- znak zapytania (?) użyty w nazwie zastępuje dokładnie jeden dowolny znak, np. prog? – może oznaczać dowolną z nazw: prog1, prog5, progi itp.,
- nawiasy kwadratowe [ ] użyte w nazwie zastępują dokładnie jeden znak spośród wyliczonych wewnątrz tych nawiasów, np.: prog [i15] oznacza dowolną z nazw – progi, prog1, prog5; ciąg kolejnych znaków w nawiasach można również przedstawić skrótowo posługując się myślnikiem, np.: prog [1-5] oznacza nazwy prog1, prog2, prog3, prog4, prog5.

PRAWA DOSTĘPU

Ważną właściwością każdego pliku są tzw. prawa dostępu (ang. access permissions), stwierdzające czy określony użytkownik jest uprawniony do odczytania lub zapisania pliku, lub do jego wykonania, jeżeli jest to program lub plik poleceń. W przypadku plików będących skorowidzami, wykonanie rozumie się jako możliwość dostępu do zawartości plików znajdujących się w wykazie, o ile ich własne atrybuty nie zabraniają takiego dostępu. Wymienione prawa, zwane też atrybutami, mogą być zróżnicowane w stosunku do każdego z trzech rodzajów użytkowników: właściciela, członków pewnej grupy i wszystkich, tak że każdemu plikowi jest przyporządkowane dziewięć bitów określających jego prawa dostępu. Dodatkowy bit, często występujący łącznie z bitami dostępu, choć nie mający z nimi nic wspólnego, określa czy plik jest skorowidzem.

W praktyce więc, prawa dostępu do pliku określa dziewięć bitów, z których trzy (na pozycjach 2-4) oznaczają kolejno: prawo do odczytu *r*, zapisu *w* i wykonania *x* dla właściciela pliku, podobnie trzy bity (na pozycjach 5-7) oznaczają prawo do odczytu, zapisu i wykonania dla grupy użytkowników, a trzy pozostałe (8-10) – odpowiednie prawa dla wszystkich (pierwszy bit oznacza, jak już wspomniano czy plik jest skorowidzem).

Prawo do wykonania wszystkich czynności dla wszystkich użytkowników oznacza się więc jako:

-rwxrwxrwx

Jednakże taka sytuacja zachodzi dość rzadko. Na ogół niektórzy użytkownicy nie są uprawnieni do wykonania niektórych operacji. Przykładowo, pliki ze skorowidza pierwotnego (root), czyli inne skorowidze, mają zazwyczaj atrybuty

drwxr-xr-x

tzn. są zapisywalne tylko przez właściciela, a pliki ze skorowidza bin, tj. programy usługowe, mogą mieć następujące atrybuty:

-r-xr-xr-x

co oznacza, że nie można w ogóle zmienić ich zawartości (jest to celowe, aby uchronić się, na przykład, przed przypadkowym i niezamierzonym zniszczeniem jakiegoś pliku).

Pliki prywatne, tzn. niedostępne dla innych, poza właścicielem, powinny mieć następujące atrybuty

-rx-------

choć na jądro systemu, tzn. na plik *unix* nakłada się mniejsze restrykcje:

-rwxr--r--

podobnie jak na pliki ze skorowidza *etc*, zawierającego na ogół istotne parametry systemu:

-rw-r--r--

Bezpośrednio po utworzeniu plik otrzymuje zazwyczaj domyślne prawa dostępu, np.:

-rwxr-x--x

co oznacza pełne prawa dostępu dla właściciela, prawo do odczytu i wykonania dla grupy i jedynie prawo do wykonywania dla wszystkich.

DOŁĄCZANIE PLIKÓW

W pracy z każdym komputerem występuje potrzeba korzystania z pojedynczych plików innych użytkowników i z całych jednostek pamięci masowej, w której są zapisane pliki własne lub innych użytkowników. Są to więc dwa aspekty tego samego problemu. Z pojedynczych plików korzysta się najczęściej tworząc do nich *łączniki* (ang. links), natomiast innych jednostek pamięci masowej używa się dołączając w całości ich systemy plików.

Łączniki do plików tworzy się po to, aby uniknąć niepotrzebnego kopiowania i przechowywania wielokrotnych kopii tego samego pliku. W przypadku utworzenia łącznika specjalnym poleceniem, plik otrzymuje drugą nazwę (traktowaną jako synonim), lecz jego właścicielem pozostaje poprzedni użytkownik. Przy tworzeniu łączników obowiązują pewne ograniczenia, np. nie można utworzyć łącznika do skorowidza i nie można zmienić praw dostępu do pliku przez łącznik (może to uczynić tylko pierwotny właściciel).

Dołączanie nowych jednostek pamięci masowej, tzw. usuwalnych (ang. removable), np. taśm magnetycznych lub dyskietek, do jednostek zainstalowanych permanentnie, jest równie częste jak dołączanie pojedynczych plików. W takiej sytuacji należy, po pierwsze, zapewnić istnienie hierarchicznego systemu plików w dołączanej jednostce (wystarczy istnienie jedynie skorowidza pierwotnego), a po drugie, wprowadzić skorowidz pierwotny tego systemu plików na miejsce jednego z pustych skorowidzów systemu podstawowego. Do tego celu służą odpowiednie polecenia systemowe. Należy jednakże pamiętać, że do zwykłych plików znajdujących się na jednostkach usuwalnych pamięci masowej na ogół nie można utworzyć łącznika od innych użytkowników, ponieważ w przypadku fizycznego odłączenia tej jednostki nastąpiłoby faktyczne usunięcie plików, natomiast w skorowidzu nadal figurowałyby ich łączniki.

PLIKI STANDARDOWE

Istotną cechą systemu jest istnienie trzech standardowych plików specjalnych, tj. standardowego pliku wejściowego, będącego domyślnym źródłem informacji dla systemu (zwykle jest to klawiatura terminala), standardowego pliku wyjściowego, będącego domyślnym miejscem przeznaczenia informacji pochodzącej z systemu (może to być np. ekran monitora), i standardowego pliku przyjmującego meldunki o błędach (pliku diagnostycznego).

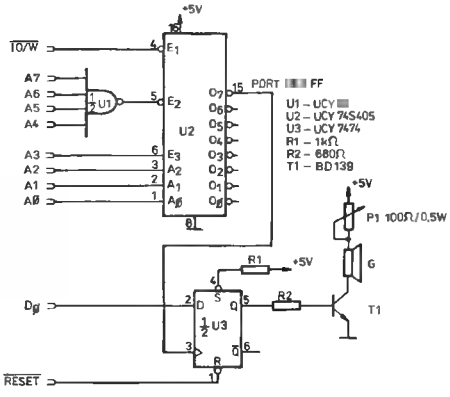
cdn.

JANUSZ ZALEWSKI

KATARYNKA

Tym, którzy nie spędzają (na razie) całych dni na instalacji dyskowych systemów operacyjnych lub konstruowaniu systemów wieloprocesorowych, proponujemy mikroprocesorową katarynkę, która przypominałaby w zimowe wieczory przeboje minionego lata. Bywalcy filharmonii być może nie będą w pełni usatysfakcjonowani, ale doświadczenia zdobyte przy konstruowaniu katarynki mogą się jeszcze przydać.

Potrzebny będzie najprostszy mikrokomputer zbudowany w oparciu o mikroprocesor Intel 8080 lub Zilog Z80, wyposażony w port wyjściowy. Ponieważ wykorzystywać będziemy tylko jeden bit tego portu, może on ograniczyć się do układu przerzutnika z dekoderem adresowym. Przykład takiego układu znajdziemy na rysunku 1. Stan linii D_0 szyny danych zapisywany jest do przerzutnika narastającym zboczem ujemnego impulsu pochodzącego z dekodera. Dekoder generuje impuls, jeżeli na liniach adresowych $A_0...A_7$ występuje poziom wysoki (adres FFH) i mikroprocesor zapisuje dane do układów we-wy (linia IOW/ma poziom niski). Elementem wykonawczym naszej katarynki będzie miniaturowy głośnik, sterowany z wyjścia przerzutnika poprzez tranzystor średniej mocy. Działaniem układu steruje program napisany w języku Assemblera µP 8080.



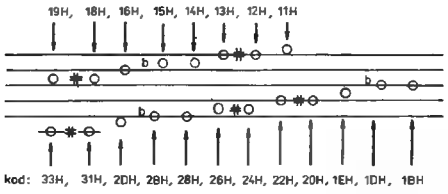
Rys. 1. Schemat ideowy portu katarynki

Program ten ma zadanie generować na wyjściu portu przebieg prostokątny o zadanych parametrach czasowych. O wysokości dźwięku, jaki popłynie z głośnika oczywiście decydować będzie częstotliwość przebiegu. Aby dźwięki ułożyły się w lubianą przez nas melodię, będziemy musieli ustalić jak długo ma trwać każdy dźwięk.

W naszej katarynce każdą nutę opisywać będą dwa bajty danych. Pierwszy bajt określa wysokość dźwięku, drugi – czas jego trwania. Ciąg takich par bajtów, który tworzyć będzie blok danych dla zasadniczego programu – to nasza mikrokomputerowa partytura.

Zasadniczy program składa się z dwu części. Pierwszą z nich jest pętla, której zadaniem jest pobieranie danych, a drugą podprogram *SOUND* – odpowiedzialny za sterowanie portu, zgodnie z aktualnymi danymi. Program umieszczony jest w pamięci począwszy od adresu [illegible] i zajmuje 79 komórek. Dane pobierane są z obszaru rozpoczynającego się od adresu 8250H.

Jak już wspomnieliśmy, dane zorganizowane są w postaci ciągu par bajtów. Pierwszy bajt każdej pary określa wysokość dźwięku zgodnie z rys. 2. Drugi bajt określa długość dźwięku, liczoną w wielokrotności podstawowego *taktu* katarynki. Takt zależny jest od częstotliwości zegara systemu, możemy jednak przyjąć, że odpowiada ósemce w średnim tempie muzycznym. Ponadto w pierwszym bajcie wystąpić mogą dwie wartości o specjalnym znaczeniu. Bajt [illegible]H powoduje powtórne nastawienie wskaźnika danych na początek bloku i w konsekwencji powtórzenie melodii. Bajt, w którym najbardziej znaczący bit będzie miał wartość *1*, czyli szesnastkowo od [illegible]H do FFH oznacza pauzę; trwa ona tyle taktów katarynki, ile żąda tego drugi parametr opisu nuty.

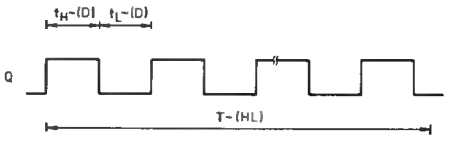


Rys. 2. Oznaczenia kodowe dźwięków katarynki

Program rozpoczyna działanie od zapisania do pary rejestrów HL adresu początku bloku danych (linie 7). Po wprowadzeniu do akumulatora danej z komórki wskazywanej przez parę rejestrów HL (linia 8) sprawdzamy, czy wczytany bajt równy jest zero (linia 9), jeżeli tak – następuje przejście do początku programu. W przeciwnym razie pobieramy następny bajt danych do rejestru *C*, a czekający dotąd w akumulatorze pierwszy bajt danych przepisujemy do rejestru *B* (linie 10-13). Mamy teraz parametr określający wysokość dźwięku w rejestrze *B* i czas jego trwania w rejestrze *C*. Sprawdzamy, czy bit b_7 w akumulatorze równy jest 1 (linia 14). Jeżeli tak, czyli gdy po wykonaniu rozkazu RLC ustawiony jest bit przeniesienia w rejestrze warunków, to przechodzimy do rozkazu opatrzonego etykietą *WAIT*. Jeżeli nie – to wykonywana jest pętla generowania dźwięku (linie 16-20). Przedtem para rejestrów *HL* zostaje odłożona na stos, gdyż będzie nam potrzebna w podprogramie *SOUND*, a przecież musimy wiedzieć, skąd pobrać parametry następnej nuty. Następnie wywołujemy procedurę *SOUND* tyle razy, ile wynosi zawartość rejestru *C*. Po wyzerowaniu tego rejestru jesteśmy już gotowi do generowania następnego dźwięku. Wystarczy tylko wskazać jego parametry parą rejestrów *HL* (linia 21) i skoczyć do drugiej linijki programu.

Podobnie działa program *generujący* pauzę. Po zapamiętaniu starej zawartości rejestrów *HL* na stosie zmniejszamy jej zawartość od początkowej wartości 50FFH tak długo, aż rejestr *H* wyzeruje się (linie 45-50). Odpowiada to czasowi podstawowego taktu katarynki. Ponieważ długość pauzy, określona liczbą taktów wynika z zawartości rejestru *C* – powtarzamy tę operację aż do wyzerowania tego rejestru (linie 51, 52). Dalej odtwarzamy parę rejestrów *HL* i już możemy zabrać się za następny dźwięk (linie 53-55).

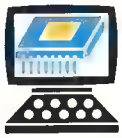
Podprogram SOUND generuje na wyjściu portu falę prostokątną. Parametr określający wysokość dźwięku przepisywany jest najpierw z rejestru *B* do rejestru *D*. Następnie wysyłamy do portu wartość logiczną 1 i w pętli zmniejszamy zawartość rejestru *D* (linie 24-33). Po wyzerowaniu rejestru *D*, znów przepisujemy do niego zawartość rejestru *B* i dekrementujemy go (zmniejszamy), tym razem utrzymując na wyjściu portu stan logiczny 0 (linie 34-42). Następnie powtarzamy obie pętle uzyskując na wyjściu falę prostokątną o żądanej częstotliwości. Pozostaje określenie momentu, w którym należy przerwać tę operację. Zwróćmy uwagę, że nie wchodzi tu w rachubę zliczanie liczby okresów przebiegu wyjściowego, ponieważ podstawowy takt byłby inny dla różnych wysokości dźwięku! Pozostaje inna droga. Otóż każdej dekrementacji rejestru *D* towarzyszy zmniejszenie zawartości pary rejestrów *HL*, do której wstępnie wpisywana jest wartość 35FFH. Po każdej dekrementacji pary rejestrów *HL* następuje sprawdzenie, czy rejestr *H* wyzerował się (linie 24 oraz 29-31 i 38-40). Jeżeli tak, to następuje powrót do zasadniczej pętli programu (linie 31 i 40). Może to nastąpić zarówno w czasie, gdy wyjście portu jest w stanie 1 jak i 0 – ale zawsze po tej samej liczbie taktów zegarowych systemu – czyli po takim samym czasie. Poglądowo przedstawia to rys. 3.



Rys. 3. Zależności czasowe na wyjściu przerzutnika U3

Obok programu zamieszczamy *partyturę* przykładowej melodyjki. Czytelnicy sami ocenią, czy melodyjka ta *wpada w ucho*. Jednocześnie zapraszamy do nadsyłania mikroprocesorowych partytur znanych przebojów. A może tak popularne zeszłego lata *Chałupy* ?...

Przygotował
JERZY ORKISZEWSKI
na podstawie materiałów firmy NEC



JAK DOŁĄCZYĆ 2048 PORTÓW DO ZX SPECTRUM

```
1:                      ;       PROGRAM KATARYNKA
2:                      ;
3:  00FF =              PORT    EQU     0FFH
4:                      ;
5:  8200                        ORG     8200H
6:                      ;
7:  8200 215082         START:  LXI     H,SONG
8:  8203 7E             START1: MOV     A,M
9:  8204 A7                     ANA     A
10: 8205 CA0082                 JZ      START
11: 8208 23                     INX     H
12: 8209 4E                     MOV     C,M
13: 820A 47                     MOV     B,A
14: 820B 07                     RLC
15: 820C DA1382                 JC      WAIT
16: 820F E5             LOOP:   PUSH    H
17: 8210 CD1182                 CALL    SOUND
18: 8213 E1                     POP     H
19: 8214 0D                     DCR     C
20: 8215 C20F82                 JNZ     LOOP
21: 8218 23                     INX     H
22: 8219 C30382                 JMP     START1
23:                     ;
24: 821C 21FF35         SOUND:  LXI     H,35FFH
25: 821F 05             SOUND1: MOV     D,B
26: 8220 3E01                   MVI     A,1
27: 8222 D3FF                   OUT     PORT
28: 8224 2B             SOUND2: DCX     H
29: 8225 7C                     MOV     A,H
30: 8226 A7                     ANA     A
31: 8227 00                     NOP
32: 8228 15                     DCR     D
33: 8229 C22482                 JNZ     SOUND2
34: 822C 50                     MOV     D,B
35: 822D 3E00                   MVI     A,0
36: 822F D3FF                   OUT     PORT
37: 8231 2B             SOUND3: DCX     H
38: 8232 7C                     MOV     A,H
39: 8233 A7                     ANA     A
40: 8234 C8                     RZ
41: 8235 15                     DCR     D
42: 8236 C23182                 JNZ     SOUND3
43: 8239 C31F82                 JMP     SOUND1
44:                     ;
45: 823C D5             WAIT:   PUSH    D
46: 823D 21FF50         WAIT1:  LXI     H,50FFH
47: 8240 2B             WAIT2:  DCX     H
48: 8241 7C                     MOV     A,H
49: 8242 B7                     ANA     A
50: 8243 C24082                 JNZ     WAIT2
51: 8246 0D                     DCR     C
52: 8247 C23D82                 JNZ     WAIT1
53: 824A E1                     POP     H
54: 824B 23                     INX     H
55: 824C C30382                 JMP     START1
56:                     ;
```

Wydruk 1. Program katarynki

```
57: 8250                        ORG     8250H
58:                     ;
59: 8250 33             SONG:   DB      33H
60: 8251 03                     DB      03H
61: 8252 2D012B03               DB      2DH,01H,2BH,03H
62: 8256 33012B02               DB      33H,01H,2BH,02H
63: 825A 33022B04               DB      33H,02H,2BH,04H
64: 825E 2D032B01               DB      2DH,03H,2BH,01H
65: 8262 26022B01               DB      26H,02H,2BH,01H
66: 8266 2D012608               DB      2DH,01H,26H,08H
67: 826A 2B032601               DB      2BH,03H,26H,01H
68: 826E 22032B01               DB      22H,03H,2BH,01H
69: 8272 22022B02               DB      22H,02H,2BH,02H
70: 8276 22042603               DB      22H,04H,26H,03H
71: 827A 22011E02               DB      22H,01H,1EH,02H
72: 827E 22012601               DB      22H,01H,26H,01H
73: 8282 1E082203               DB      1EH,08H,22H,03H
74: 8286 33012D01               DB      33H,01H,2DH,01H
75: 828A 28012601               DB      28H,01H,26H,01H
76: 828E 22011E08               DB      22H,01H,1EH,08H
77: 8292 1E032D01               DB      1EH,03H,2DH,01H
78: 8296 28012601               DB      28H,01H,26H,01H
79: 829A 22011E01               DB      22H,01H,1EH,01H
80: 829E 1B081B03               DB      1BH,08H,1BH,03H
81: 82A2 28012601               DB      28H,01H,26H,01H
82: 82A6 22011E01               DB      22H,01H,1EH,01H
83: 82AA 1B011907               DB      1BH,01H,19H,07H
84: 82AE 1B011E02               DB      1BH,01H,1EH,02H
85: 82B2 26021B02               DB      26H,02H,1BH,02H
86: 82B6 22021906               DB      22H,02H,19H,06H
87: 82BA FF08                   DB      0FFH,08H
88: 82BC 00                     DB      0
89:                     ;
90: 82BD                        END
```

Wydruk 2. Blok danych do programu – przykładowa melodia

Bardzo często mikrokomputer ZX Spectrum jest uważany za zabawkę służącą jedynie do uruchamiania gier. Natomiast bardzo rzadko myśli się o nim jako o komputerze, który może być centralną częścią złożonego systemu zawierającego wiele dodatkowych urządzeń zewnętrznych. Jedną z przyczyn takiej sytuacji jest opinia, powstała zresztą w wyniku nie najlepszego opisu firmowego, że ZX Spectrum może zaadresować zaledwie 8 portów we-wy. W rzeczywistości istnieje kilka sposobów umożliwiających zwiększenie tej liczby nawet do 2048. Oznacza to, że systemy wykorzystujące ZX Spectrum mogą być praktycznie dowolnie rozbudowywane. Ponieważ jeden z przedstawionych poniżej sposobów wykorzystuje niezbyt dobrze znane właściwości mikroprocesora Z80, to wydaje się, że niektóre uwagi mogą zainteresować także użytkowników innych mikrokomputerów opartych na tym mikroprocesorze.

Obsługa urządzeń zewnętrznych przez ZX Spectrum może być dokonana za pomocą 12 instrukcji języka assemblera Z80 oraz 2 instrukcji języka BASIC (**OUT adres, n oraz IN adres**, gdzie $0 \leqslant$ adres $< 65536, -255 \leqslant n \leqslant 255$). Wprawdzie szyna adresowa mikroprocesora Z80 zawiera 16 linii, jednak, jak wynika z 23 rozdziału podręcznika języka BASIC [1], liczba linii dostępnych dla użytkownika jest znacznie ograniczona. Przede wszystkim linie od A0 do A4 są używane do adresowania wewnętrznego sterownika ULA oraz przewidzianych przez producenta standardowych urządzeń zewnętrznych: drukarki, napędów mikrokaset (ang. microdrive) i interfejsu RS 232 C. Każda z tych linii wybiera jeden port i dlatego w czasie operacji we-wy tylko jedna z nich może być w stanie niskim.

Poza tym, jak podano w [5] „linie A8, A9 i dalsze są czasem używane do przesyłania dodatkowych informacji". Wynika stąd, że do wykorzystania pozostają tylko linie adresowe A5, A6 i A7, co pozwala na zaadresowanie zaledwie 8 portów (jak się później okaże nie jest to wcale prawda). Oczywiście najprostszą metodą rozszerzenia przestrzeni adresowej portów we-wy jest (na przekór zaleceniom producenta) użycie jednej z aktualnie niewykorzystywanych linii A1...A4. Jeśli przykładowo, w skład systemu nie wchodzi drukarka, to możliwe jest użycie linii A2 do zaadresowania 8 dodatkowych portów. Należy się jednak liczyć z możliwością przykrych konsekwencji takiego postępowania. Jeżeli bowiem, nawet przypadkowo, zostanie użyta jedna z instrukcji obsługi drukarki (np. COPY) wówczas może dojść do błędnej obsługi portu o adresie 251. Poza tym mogą powstać problemy w wypadku przyłączenia drukarki do komputera. Można wówczas przeznaczyć dla drukarki adres 251 i zbudować dekoder tego adresu. W ten sposób pozostanie i tak 15 wolnych portów we-wy, co oznacza pewien zysk w porównaniu z sytuacją początkową.

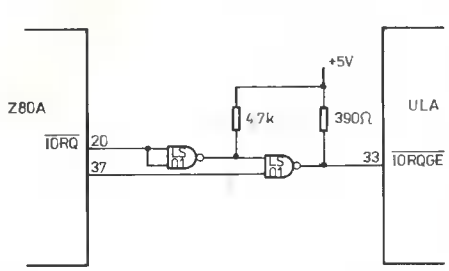
Inny sposób został opisany w [1]. Pozwala on na dołączenie 128 portów we-wy, jednak wymaga dokonania zmiany wewnątrz komputera. W metodzie tej zakłada się, że adresy wszystkich dodatkowych urządzeń zewnętrznych mają A7=0. Idea polega tu na blokowaniu drogi sygnału */ORQ/* (Input Output Request) do standardowych urządzeń zewnętrznych w czasie gdy A7=0. Normalnie wyjście */ORQ*/mikroprocesora Z80 jest połączone z wejściem */ORQGE*/układu ULA za pomocą rezystora 680 Ω. Po dokonaniu modyfikacji, zgodnie z rysunkiem 1, sygnał */ORQ/* dotrze do układu ULA tylko wtedy, gdy A7=1. Zmodyfikowany sygnał */ORQGE/* należy także doprowadzić do pozostałych standardowych urządzeń zewnętrznych zamiast sygnału */ORQ/* (oczywiście dodatkowe urządzenia zewnętrzne powinny być połączone z */ORQ/*). Jeżeli wystarczą tylko 64 porty we-wy i nie chcemy dokonywać zmian wewnątrz komputera można zrezygnować z modyfikacji i połączenia między mikroprocesorem Z80 i układem ULA. Ze zmiany tej należy również zrezygnować w tych egzemplarzach ZX Spectrum, w których do linii *A0* i */ORQGE/* jest przylutowany tranzystor.

Następny sposób umożliwia dołączenie 256 portów we-wy jednak powoduje pewną komplikację układu elektronicznego i programu obsługi. Polega on na wysyłaniu instrukcji we-wy w dwóch etapach z wykorzystaniem dwóch adresów, np. 127 i 255. W pierwszym etapie adres portu, który ma być obsłużony, zostaje wysłany pod adres 127 i zapamiętany w buforze. W drugim etapie następuje przesłanie danych przez port o adresie 255.

Dodatkową wadą tego sposobu jest zmniejszenie szybkości przesyłania danych z (do) urządzeń zewnętrznych.

Ostatni sposób wykorzystuje tę właściwość mikroprocesora Z80, że do adresowania układów we-wy można wykorzystać w pewnych sytuacjach nie 8 linii adresowych (tak jak w mikroprocesorze 8080), lecz 16!. Ponieważ jest to możliwość chyba niezbyt dobrze znana (w [3] i [4] nie wspomina się o niej) wydaje się, że warto nieco dokładniej opisać sposób wysyłania adresu przez mikroprocesor Z80 w czasie operacji we-wy.

Adresowanie portów we/wy przez mikroprocesor Z80 może być bezpośrednie lub



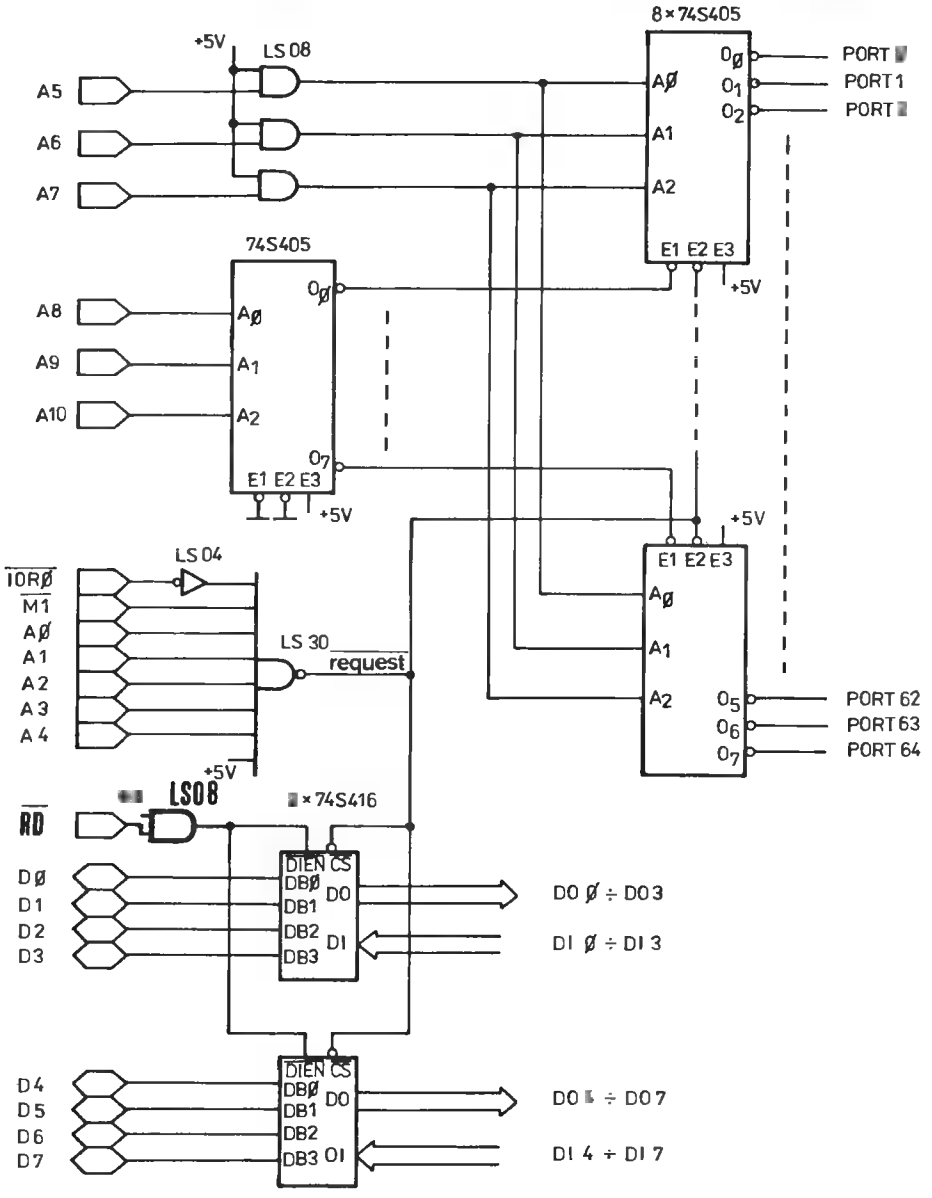
Rys. 1. Blokowanie sygnału IORQ/ mikroprocesora za pomocą sygnału A7

pośrednie. Do pierwszej grupy należą dwie instrukcje **IN A,(n)** i **OUT (n),A**, które są odpowiednikami instrukcji **IN** r oraz **OUT** r występującymi w mikroprocesorze 8080. Adres *n* może przybierać wartości od 0 do 255. W czasie wykonywania tych instrukcji adres *n* jest wysyłany na linie adresowe A0...A7, natomiast na liniach A8...A15 pojawia się zawartość akumulatora (w mikroprocesorze 8080 na liniach A8...A15 pojawia się jeszcze raz adres *n*). Do adresowania portów we-wy w sposób pośredni wykorzystuje się rejestr C. Do grupy tej należą instrukcje **IN r,(C), OUT (C),r** oraz osiem innych, służących do blokowego przesyłania danych między pamięcią i urządzeniami zewnętrznymi [3, 4]. Instrukcja **IN r,(C)** oznacza „wprowadź daną z urządzenia zewnętrznego o adresie określonym przez zawartość rejestru C do rejestru roboczego *r*, gdzie *r* może być równe A, B, C, D, E, H lub L", natomiast instrukcja **OUT (C),r** oznacza „wyślij daną z rejestru roboczego *r* do urządzenia zewnętrznego o adresie określonym przez zawartość rejestru C". Instrukcje blokowego przesyłania danych z adresowaniem pośrednim przez zawartość rejestru C wykorzystują dodatkowo rejestr B, który służy jako licznik przesłanych bajtów. W czasie wykonywania instrukcji we-wy z adresowaniem pośrednim, zawartość rejestru C pojawia się na liniach adresowych A0...A7, natomiast zawartość rejestru B na liniach A8...A15 (!). Wynika z tego, że w instrukcjach **IN r,(C)** oraz **OUT (C),r** rejestr B może być wykorzystany do rozszerzenia pola adresowego układów we-wy. W tym celu należy załadować do rejestru BC adres urządzenia zewnętrznego *n* (przy czym $0 \leqslant n < 65536$) oraz wysłać daną instrukcję. Ponieważ bity A0...A4 nie powinny być w ZX Spectrum wykorzystane, użytkownik ma w rzeczywistości do dyspozycji 11 linii adresowych od A5 do A15, co pozwala na zaadresowanie 2048 portów zewnętrznych.

Praktycznych układów interfejsu między mikrokomputerem a urządzeniami zewnętrznymi, wykorzystujących wyżej opisaną właściwość mikroprocesora Z80, może być wiele. Przykładowy interfejs, pozwalający na dołączenie 64 portów we-wy jest przedstawiony na rysunku 2. Odwołanie mikroprocesora do dodatkowych urządzeń zewnętrznych jest dekodowane w prosty sposób za pomocą bramki NAND 74LS30. Sygnał *REQUEST/* na wyjściu tego układu pojawia się, gdy mikroprocesor wysyła sygnał *IORQ/*, nie wysyła w tym czasie sygnału *M1* oraz gdy nie jest wybrane żadne standardowe urządzenie zewnętrzne (tzn. stan na liniach A0...A4 jest wysoki). Stan linii M1 jest ważny, gdyż sygnał *IORQ/* jest również generowany wraz z sygnałem M1 w cyklu obsługi przerwania w celu zasygnalizowania, że na magistralę danych można przesłać wektor adresu przerwań.

Do adresowania urządzeń zewnętrznych są wykorzystane linie A5...A10. Adres jest dekodowany za pomocą kaskadowo połączonych układów 74S405. Na podstawie stanu linii A8, A9 i A10 wybierany jest jeden z ośmiu dekoderów wyjściowych. Dekodery te są sterowane mniej znaczącymi bitami A5, A6 oraz A7 i uaktywniane sygnałem *REQUEST/*. Linie adresowe A5...A7 są buforowane za pomocą bramek LS08 w celu obniżenia prądu pobieranego z mikroprocesora.

Dane są przesyłane przez dwa układy 74S416 (nadajnik/odbiornik szyny danych). Układy te są uaktywniane sygnałem



Rys. 2. Interfejs umożliwiający podłączenie 64 portów we-wy do ZX Spectrum

REQUEST/. Kierunek przesyłania danych jest wybierany za pomocą sygnału *RD/* mikroprocesora. Dane zapisywane do portów zewnętrznych są dostępne na wyjściach D00...D07, natomiast dane czytane przez mikroprocesor powinny pojawiać się na wejściach DI0...DI7.

Programowanie interfejsu jest bardzo proste, gdyż instrukcje w BASICu służące do obsługi urządzeń zewnętrznych wykorzystują właśnie dwie omawiane instrukcje Assemblera Z80 **IN r,(C)** i **OUT (C),r**. (Adresy w ROM ZX Spectrum: 34A8H i 1E7DH). Przykładowy program wysłania liczby 128 do portu nr 10 jest przedstawiony poniżej.

```
10 DEF FN a(n) = n x 32 + 31
...
100 OUT FN a(10), 128
```

Pewną wadą przedstawionego rozwiązania jest to, że w programach pisanych w języku Assembler Z80 nie można wykorzystywać pozostałych instrukcji we-wy, tj. instrukcji z adresowaniem bezpośrednim i instrukcji blokowego przesyłania danych. Nie jest to jednak wada zbyt duża, gdyż każda z tych instrukcji może być zastąpiona przez krótki program wykorzystujący tylko te dwie podstawowe instrukcje.

Literatura

[1] Dickens A.: Spectrum Hardware Manual, Melbourne House, 1983
[2] Logan I., O'Hara F.: ROM Disassembly Book, Melbourne House, 1983
[3] Misiurewicz P.: Układy mikroprocesorowe, WNT, 1983
[4] Modułowe systemy mikrokomputerowe, WNT, 1984
[5] Vickers S.: Sinclair ZX Spectrum BASIC Programming, Cambridge, 1982.

KRZYSZTOF KONTEK
IPJ Świerk

ZAŚLEPIENIE KONTRA ZACOFANIE

Mikrokomputerowa moda powoduje, że coraz więcej mówi się o konieczności wprowadzenia nowej techniki do procesu nauczania w polskich szkołach. Niestety, znaczna część dyskutantów rzadko dostrzega coś więcej niż problem zdobycia odpowiedniej ilości sprzętu. Używane są często demagogiczne argumenty, takie jak „cywilizacyjna konieczność", „groźba powrotu do epoki kamienia łupanego" itp., nie prowadzące do rzeczywiście racjonalnych rozwiązań, natomiast gorąca atmosfera dyskusji stwarza niebezpieczeństwo wymuszenia na czynnikach decyzyjnych niedostatecznie przemyślanych decyzji. Z tych to właśnie powodów zamierzamy tej niezwykle ważnej problematyce poświęcić sporo miejsca publikując doświadczenia zagraniczne.

Dyskusję o edukacji wspomaganej przez komputery celowo rozpoczynamy refleksjami z prasy amerykańskiej na temat pewnych negatywnych aspektów wprowadzania komputerów do szkół w USA. Analizując treść tej wypowiedzi należy zauważyć, że pojawiła się ona w kraju, gdzie w wyniku wieloletniego i całkowicie spontanicznego rozwoju musiały już wystąpić również zjawiska negatywne, towarzyszące każdej działalności człowieka. Poza tym autor ujawnia swój wyłącznie humanistyczny punkt widzenia, nie dostrzegając w komputerze również cech pozytywnych. Sądzimy, że takie, trochę prowokujące większość naszych Czytelników, poglądy wywołają z pewnością rezonans, z którego wyłonią się dojrzałe i przemyślane wypowiedzi na temat racjonalnego wykorzystania mikrokomputerów w szkołach. (Red.)

Amerykańscy nauczyciele są tak zachwyceni komputerem, że stają się wobec niego coraz bardziej bezkrytyczni. Ogłaszając „rewolucję komputerową w nauczaniu" nie zadają sobie trudu by stwierdzić, na jakim poziomie, dla jakich celów i jak wykorzystywany komputer jest przydatny bądź nieprzydatny w nauczaniu; kiedy i dla kogo jest pożyteczny, a w jakich sytuacjach przynosi więcej strat niż korzyści.

Nauczyciele prześcigają się w wymyślaniu nowych zastosowań komputerów wszędzie tam, gdzie istnieje cień możliwości i – w rezultacie – przestają kierować się poczuciem odpowiedzialności zawodowej. Jakie są tego przyczyny? Głównie względy finansowe (tam, gdzie jest komputer – są pieniądze) i lęk przed utratą pozycji zawodowej (lepiej być rzecznikiem nowych, zwycięskich idei niż się im przeciwstawiać). I jeszcze jedno: gdy amerykańskie szkoły państwowe zaczęły borykać się z trudnościami, komputer pojawił się deux ex machina, by uzdrowiać sytuację i zmusić nauczycieli, rodziców i urzędników do wnikliwej rewizji poglądów w stosunku do siebie oraz społeczeństwa.

Kariera komputerów w świecie dorosłych stwarza pokusę, by drzwi do „nowego świata" otworzyć również dzieciom, czyli jak najszybciej zrobić z nich dorosłych. Pojawiły się obawy – podsycane przez przemysł komputerowy i szkoły – że dzieci, które nie zapoznają się z komputerem od kołyski, na zawsze stracą swoje życiowe szanse. Nie było warunków, by rozsądnie pomyśleć co jest rzeczywiście najlepsze dla młodzieży i jej postępów w nauce. Nie trzeba być fanatycznym przeciwnikiem postępu, aby zastanawiać się, czy aby na pewno firmy komputerowe i współpracujące z nimi wydawnictwa, organizujące wielkie kampanie wprowadzania setek tysięcy komputerów do szkół, były zainteresowane tylko podstawowymi potrzebami młodych ludzi? Pytanie: kto naprawdę korzysta z rewolucji komputerowej, pozostaje nadal bez odpowiedzi.

Podstawową przyczyną porażki nauczycieli w próbach określenia perspektyw wykorzystania komputerów w szkolnictwie jest brak odpowiedzi na pytania dotyczące istoty samego procesu nauczania. Literatura fachowa poświęca bardzo mało miejsca sprzeczności pomiędzy ogólnie uznawanymi celami nowoczesnej edukacji a zastosowaniem komputera. Coraz wyraźniej rysuje się natomiast zbieżność przeświadczeń nauczycieli, rodziców i większości społeczeństwa, że głównym celem nauczania jest rozwój umiejętności rozwiązywania problemów technicznych. Dokładniej – technicznych, użytecznych, logicznych – a więc takich, do których najlepiej nadaje się komputer. Zapomina się, że prawdziwe poznanie wymaga większej intuicji i wyobraźni niż rozstrzyganie zagadnień czysto technicznych. Zmuszanie małego dziecka do analitycznego i abstrakcyjnego myślenia – w cieplarnianych warunkach – nie zapewni mu zdobycia umiejętności logicznego rozumowania. Nie zwraca się uwagi, że to co dobre dla studentów i dorosłych może być wręcz szkodliwe dla dzieci i młodzieży szkolnej. Nic dziwnego, że nawoływanie w opisanej sytuacji do krytycznego spojrzenia na komputeryzację jest dla wielu wyrazem postawy i myślenia antytechnicznego. A przecież główny problem to nie postawa popierająca lub sprzeciwiająca się stosowaniu komputerów w szkolnictwie, ale stworzenie warunków, które umożliwiałyby prawdziwie humanistyczne wykorzystanie maszyn.

W godnym tego miana procesie nauczania przełomowym momentem jest uwzględnienie głównej roli, jaką odgrywa w nim czynnik emocjonalny. Emocje kierują i uzupełniają logiczne myślenie, pozwalają określać cele i dostarczają energii do ich osiągnięcia. I – co najważniejsze – gdy są prawidłowo rozwijane i kształtowane – stają się najbardziej wnikliwym i niezastąpionym środkiem poznania. Tylko dzięki nim można poznać różnorodne wartości życia – w naturze, w innych ludziach, w sobie.

Sama wrażliwość to jednak jeszcze zbyt mało. Fałszywe racje rodzą fałszywe emocje. Niezbędne jest kształcenie i rozwijanie dyscypliny wewnętrznej, umiejętności właściwego wykorzystania energii, dokonywania wyborów i wartościowania. Współczesna pedagogika jakby przestała rozumieć znaczenie świata emocji. Niektórzy psychologowie-pedagodzy popierają nauczanie odwołujące się do emocji, ale też prawie zawsze przeciwstawiają je nauczaniu odwołującemu się do rozumu, co przynosi raczej więcej szkody niż pożytku. Z drugiej strony w najnowocześniejszej pedagogice traktuje się emocje jako problem – gdy nie można ich poskromić, może uda się je wyeliminować albo wykorzystać – a więc rozwiń swoje zainteresowania, uczyń naukę przyjemną.

Komputer jest często zalecany szkołom, ponieważ zachęca dzieci do nauki. Należy jednak podkreślić, że z punktu widzenia pedagogiki motywacja nie jest argumentem za czymkolwiek. Brak motywacji na pewno jest problemem, ale jest nim również i rodzaj motywacji. Dzieci mogą być przecież motywowane przez różne czynniki – na przykład lody, seks czy kieszonkowe. Pobudzenie wyobraźni nie zawsze jest równoznaczne z rozwijaniem i kształceniem wrażliwości.

Ludzie, których wrażliwość jest systematycznie przytępiana szybko tracą zdolność rozumienia, że zdrowie społeczeństwa nie jest wynikiem zwiększenia dochodu narodo-

wego, nowych wyborów czy rozwoju przemysłu. Zanik wrażliwości to strata zdolności rozumienia wartości życia.

Prawdziwa edukacja wymaga przedstawienia dziecku bogactwa życia we wszelkich jego przejawach. Prawidłowy rozwój zapewnia środowisko dostarczające wielu doświadczeń zmysłowych: kolorów, dźwięków, zapachów, ruchu, kontaktu z naturą. Niezbędne jest współdziałanie i kontakt z innymi – opowiadania, przedstawienia teatralne, ruch, prace ręczne. Rezygnowanie z tych elementów w nauczaniu spowoduje, że społeczeństwo, choć silne i sprawne, będzie coraz bardziej filisterskie, kołtuńskie – bo pozbawione prawdziwej mądrości. Będzie ono większym zagrożeniem dla siebie i innych.

Podkreślając znaczenie umiejętności logicznego rozwiązywania zadań, jako głównego celu nauczania, zapomina się o roli wyobraźni w myśleniu. Prowadzi to do myślenia schematycznego i rutynowego. Nie ma więc miejsca na świeżość spojrzenia – niesprawdzalne i podświadome przebłyski intuicji pozwalające odrzucić powszednią logikę i skierować umysł na nowe drogi (niektórzy powołują się przy takich rozważaniach na przykłady Newtona, Einsteina, Poincare'go). To prawda, że pełne zrozumienie i przewidzenie implikacji owych „pierwszych olśnień" wymaga przetworzenia ich w języku hipotez, logiki formalnej i obliczeń. Ale logika formalna, nie negując jej wielkiego znaczenia, pełni w nauce rolę wtórną w stosunku do intuicji i wyobraźni – nie jest źródłem nowej wiedzy. Posługiwanie się wyłącznie formalizmami i myśleniem dyskursywnym prowadzi do schematyzmu i „szufladkowania" wiedzy.

Znaczenie wyobraźni w procesie myślenia skłania do bardzo ostrożnego wykorzystania komputerów w nauczaniu – jest to środek niewątpliwie skuteczny i atrakcyjny, ale też niezwykle specyficzny i zarazem ograniczony. Nawet fizycy teoretyczni posługujący się czystą matematyką ulegają głęboko ukrytym wyobrażeniom o naturze świata, który badają. I oni muszą uciekać się do wyobraźni, chcąc znaleźć eksperymentalne potwierdzenie słuszności swoich dociekań. Wyobraźnia i umiejętność korzystania z niej jest bardzo ważna, ponieważ właśnie ona określa obraz świata, który poznajemy i świata, który tworzymy dla nas samych.

Coraz powszechniejsza wizja nowoczesnej umysłowości jest nierozerwalnie związana z maszynami. Szczególny sposób myślenia wymuszany przez zastosowanie komputera jest ściśle mechanistyczny i właśnie on stał się udziałem naukowców kształtujących obraz świata większości ludzi. *Nie ma przesady w stwierdzeniu* – napisał w 1983 r. znany biolog R.C. Lewantin – *że większość naukowców po prostu nie wie, jak myśleć o świecie bez maszyny.* Nawet nauki przyrodnicze, zdominowane przez mechanistyczne myślenie, usiłują odnaleźć sekret życia w nieożywionym. Słynny biochemik Erwin Chargaff mówił o „paradoksie biochemii", która nie potrafi badać samego życia – musi ona uśmiercać organizmy, aby móc je poznawać. Mechanistyczny obraz świata dopuszcza manipulowanie życiem, co przynosi wymierne wyniki. Dyskusyjne jest jednak twierdzenie, że takie działania przybliżają zrozumienie istoty życia.

Jeżeli pozostaniemy wyłącznie pod wpływem modeli mechanistycznych – będziemy żyć w mechanistycznym świecie. Wiele zależy od tego, czy nasza wyobraźnia jest żywa i ruchliwa (a nie sztywna i rutynowa), czy jest zauważalna w naszym myśleniu i czy potrafimy ją wykorzystać z całą świadomością konsekwencji tego faktu. Jesteśmy odpowiedzialni za obraz świata, który tworzymy i staramy się zrozumieć. Rozwój zdrowego myślenia jest głównym zadaniem nauczania związanym z rozwojem twórczego, odpowiedzialnego myślenia o życiu i świecie.

Odczucia i wyobraźnia dzieci są pomijane w nauczaniu nastawionym na jak najszybszy rozwój umiejętności rozwiązywania ściśle określonych zadań. Czy nie ma niebezpieczeństwa w tym, że dzieci podporządkowane technice będą pozbawione innych doświadczeń, tak ważnych w wieku największej chłonności i plastyczności umysłu? Jaki będzie wpływ obcowania z dwuwymiarowym, krzykliwie kolorowym obrazem monitora ekranowego na rozwijający się, ale jeszcze nieukształtowany umysł, mózg i ciało młodego człowieka? Co stanie się z dziećmi, które zbyt wcześnie zamknięto w abstrakcyjnym i zubożonym świecie nauczania komputerowego? Czy będzie je stać na niezależne myślenie? Czy ich doświadczenia pozwolą na prawdziwie twórcze wykorzystanie komputera? Może dojdą do wniosku, że najważniejsze problemy ludzkości dają się skomputeryzować, a życie wymaga jedynie więcej informacji i obliczeń – nie zaś zrozumienia, współczucia, a często także wyrzeczeń? Czy nie będą one wymagały wkrótce pomocy – i to nie w nauce lecz w życiu osobistym?

To tylko niektóre refleksje. Komputery mogą przynieść niezmierne korzyści, ale ich nieprzemyślane zastosowanie obraca się przeciwko ludziom. Wnikliwe rozważanie roli komputerów w nauczaniu może być pierwszym krokiem w kierunku takiego ich wykorzystywania, by faktycznie przyniosły korzyść.

Oprac. KATARZYNA ISAAK

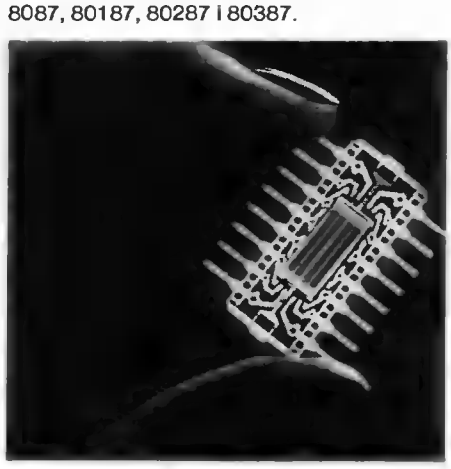
NOWE KOSTKI

Coraz większą karierę robi wprowadzona ostatnio na skalę przemysłową technologia CHMOS (oznaczenie według firmy INTEL), pozwalająca na wytwarzanie układów scalonych o charakterystycznym dla technologii CMOS niskim poborze mocy i szybkości odpowiadającej układom LS TTL. Opanowanie procesu CHMOS zaowocowało ukazaniem się na rynku półprzewodników nowych, ciekawych elementów. Firma INTEL oferuje ostatnio pamięci dynamiczne 51C259H i 51C259L. Mają one pojemność 256 Kb i organizację 64 Kb × 4. Wersja H optymalizowana jest pod względem szybkości działania, w wersji L zwrócono natomiast szczególną uwagę na mały pobór mocy. Wydaje się, że pamięci te znajdą szerokie zastosowanie w przenośnych komputerach zasilanych bateryjnie oraz w urządzeniach przemysłowych, pracujących w warunkach silnych zakłóceń. Warto bowiem dodać, że pamięci pracują w zakresie napięć zasilających i z marginesem zakłóceń typowym dla układów CMOS. Jak dotąd ich cena jest jednak bardzo wysoka – ponad 48 dol. za kostkę i to przy zamówieniach powyżej tysiąca sztuk.

★

A oto drugi przykład zastosowania technologii CHMOS, tym razem w wydaniu firmy MOTOROLA (gdzie nosi ona nazwę HCMOS). Dostępny jest już układ koprocesora arytmetycznego rodziny M68000. Koprocesor, o symbolu MC68881, przeznaczony jest do współpracy z 32 bitowym procesorem 68020, może jednak funkcjonować jako moduł peryferyjny, w systemach opartych na 16-bitowe jednostki centralne 68000 i 68010. Układ zawiera w sobie arytmometr, o długości słowa 67 bitów i osiem rejestrów 80 bitowych. Realizuje obliczenia w zmiennym przecinku, zgodnie z wymogami normy IEEE. Kostka jest odpowiednikiem 155 tys. tranzystorów, upakowanych na płatku krzemu o wymiarach 27 na 33 mm.

Warto dodać, że mianem koprocesora określamy specjalizowany procesor pracujący przemiennie z głównym procesorem systemu. Analogiczne rozwiązanie stosuje INTEL. Koprocesory arytmetyczne dla kolejnych mutacji rodziny iAPX są oznaczone: 8087, 80187, 80287 i 80387.



BIT ZA TRZY GROSZE

Rubryka BIT ZA TRZY GROSZE będzie miała nieco inny charakter niż cały MIKROKLAN. Przede wszystkim będzie ona fotografowana bezpośrednio z wydruków komputerowych. Chcielibyśmy przeznaczyć dwie strony naszej publikacji na szeroko rozumianą rozrywkę - ale nie tylko! Wśród materiałów dla hobbystów budujących własne urządzenia mikroprocesorowe oraz wielu opisów "poważnych" zastosowań i programów mikrokomputerowych nie może zabraknąć miejsca dla tych wszystkich, którzy chcieliby podzielić się swoimi pomysłami, jak inaczej używać komputery osobiste (lub jak nie używać...). Nawet jeśli pomysły te będą nieco zwariowane. Wiemy, że wielu Czytelników stosuje komputery osobiste dla celów profesjonalnych. Wiemy też, że duża grupa posiadaczy mikrokomputerów znudziła się już typowymi grami, gdzie strzela się, morduje lub tylko rozwiązuje komputerowe labirynty. Dla nich otwieramy swoje łamy BIT ZA TRZY GROSZE! Prosimy - jeśli wymyśliliście nową formę grafiki komputerowej, jeśli potraficie przy pomocy komputerka zrobić coś, czego nikt dotąd nie robił, jeśli chcecie pokazać światu, że komputer osobisty dostarcza Wam prawdziwej rozrywki - przysyłajcie materiały do naszej redakcji. Postaramy się, oczywiście w miarę naszych możliwości, zaprezentować Wasze dokonania.

Nazwa naszej rubryki nie jest przypadkowa. Chcemy wszystkie opublikowane materiały honorować według (skromnych - nie z winy redakcji) stawek mutorskich. Trzy grosze za bit informacji to dużo lub też bardzo mało. Wszystko zależy od tego, czy pracy twórczej towarzyszy prawdziwe zadowolenie. Wierzymy, że nasi Czytelnicy potrafią doskonale bawić się swoimi komputerami.

Jeszcze parę spraw organizacyjnych. Prosimy o wydruki i rysunki o jak największym kontraście. Jeśli ktoś nie ma możliwości wydrukowania swojego programu lub narysowania grafiki to niech przyśle choćby czytelny rękopis lub rysunek - takie materiały postaramy się opracować na jednym z dostępnych nam komputerów. Ze względu na format naszej publikacji pożądane wymiary materiałów powinny zmieścić się w szerokości 18.4 cm (jeszcze lepiej, gdy będą nadawać się do dwukrotnego zmniejszenia). A więc - ZAPRASZAMY!!!

Dzisiejszy, pierwszy, odcinek rubryki BIT ZA TRZY GROSZE poświęcony będzie rozważaniom jak korzystać z mikrokomputerów. Chcąc jednak zachować rozrywkowy charakter rubryki, zaczniemy od prezentacji paru rysunków, wygenerowanych przez rewelacyjny program MacPaint. Różne wersje tego programu są już dostępne dla większości mikrokomputerów tak zwanej drugiej generacji, czyli w praktyce 16 bitowych. Mamy nadzieję, że i w Polsce rozpowszechni się nowa gałąź malarstwa - malarstwa bez pędzli, farb i papieru.

Do kiosku przywieziono MIKROKLAN...

BIT ZA TRZY GROSZE

Efekt Ponzo | Efekt Kanizsa | Efekt Ehrensteina | Efekt Neokera

Wzrok daje się łatwo oszukać (niech Czytelnicy sami ocenią, czy rzeczywiście ulegają powyższym efektom!), także nasze myślenie jest zwykle bardzo sztampowe. Chętnie powielamy utarte schematy i przywiązujemy się do przekonań, których nabyliśmy w młodości. W ten sposób prawie wszyscy uwierzyliśmy, że komputery (jak to sugeruje ich nazwa!) służą do obliczeń. Rzeczywiście, w początkowym okresie, gdy komputery były duże i bardzo kosztowne, używano ich przede wszystkim do obliczeń: wojskowych (symulacja bomby atomowej), bankowych (tam pieniędzy nie brakuje), czy wreszcie naukowych (wiadomo, że naukowcy wszystko obmierzą i policzą). Na szczęście, już po trzydziestu latach, pierwsze komputery trafiły w ręce osób prywatnych. Okazało się, że mikrokomputer niezbyt nadaje się do obliczeń, gdyż jest wolny i mało dokładny. Za to świetnie można go używać do rozmaitych gier. Nadaje się też doskonale do przetwarzania tekstów (co widać na przykładzie tego artykułu, który był wielokrotnie poprawiany bez zmarnowania jednej kartki papieru!) oraz do wielu innych prac domowych i biurowych. Stąd teza pierwsza: **komputery osobiste nie służą do obliczeń**. Nie śmiem oczywiście twierdzić, że na mikrokomputerze nie wolno wykonywać obliczeń – ale rozwiązywanie równania czwartego stopnia metodą iteracyjną (i to bez zbieżności...) jest takim samym nonsensem, jak wbijanie gwoździ zegarkiem. Można, ale po co?!

Teza druga jest jeszcze bardziej przewrotna: **komputery osobiste nie służą do programowania!** Oczywiście wszyscy Czytelnicy zakrzykną: jak to! Najpierw zachęcano nas do wymyślania nowych zastosowań komputerków, by zaraz potem zabronić programowania. Chodzi jednak o rozróżnienie między programowaniem zadań nietypowych (jak grafika) a pisaniem dużych programów typu bazy danych. Te ostatnie są profesjonalnym zadaniem, które najlepiej wykonają zawodowcy. Zwyczajny użytkownik mikrokomputera powinien raczej korzystać z gotowych procedur. Granica jest tu płynna – jeśli ktoś koniecznie chce zabłysnąć jako mistrz programowania, niech raczej wymyśli nowy sposób przedstawiania informacji, wygodny dla mniej wprawnych kolegów. Albo niech wymyśli nową grę strategiczną: zawodowcy pomogą mu "wygładzić" sam program.

Po dwóch antytezach, pora na tezę konstruktywną. Twierdzę, że **komputery osobiste są wzmacniaczem kreatywności**. Znaczy to, że dzięki swej prędkości i całkowitej uległości, mikrokomputery mogą w zasadniczy sposób zmienić najbardziej intymną dziedzinę ludzkiego życia – pomogą nam tworzyć. Dlatego zachęcamy Czytelników do śmiałych prób. Pokażcie, że przy pomocy komputerka można odkryć nowy, wspaniały świat. Świat fantazji nie zmąconej ograniczaniami materialnymi, świat gdzie wszystko jest dobre, piękne i wartościowe.

Kończymy pierwszy odcinek rubryki BIT ZA TRZY GROSZE. Jeśli nie podaliśmy zbyt wielu przykładów, jak używać mikrokomputerów to dlatego, że wierzymy iż Czytelnicy zasypią redakcję swoimi pomysłami. Za to podaliśmy parę przykładów niecelowej pracy przy komputerze. Obiecujemy, że w przyszłych odcinkach będziemy bardziej pozytywni!

Mackie

Liczymy na wiele listów!!!

CENY

Pragniemy regularnie informować Czytelników o możliwie aktualnych cenach podzespołów, programów i sprzętu mikrokomputerowego (w tym również niedostępnego dla polskich użytkowników). Ceny zagraniczne podawane będą w walucie danego kraju (dolary, marki, funty), natomiast krajowe – na podstawie notowań „perskiego rynku" w Warszawie (obecnie w pobliżu skrzyżowania ulicy Obozowej i Ks. Janusza, w niedziele, godziny 7 – 12). Ceny krajowe sprzętu komputerowego podawane będą w tys zł, a zestawienie – oprócz notowań wolnorynkowych – uwzględniać będą również ceny stosowane przez przedsiębiorstwa państwowe: BOMIS – B oraz PEWEX – P (w tym ostatnim przypadku oczywiście w dolarach). Przy okazji informujemy, że przedsiębiorstwo BOMIS zajmuje się obecnie również skupem i odsprzedażą zagranicznego sprzętu komputerowego.

CENY NA DZIEŃ 1 MAJA 1986

Układy	USA	RFN	PL
	Mikroprocesory		
8080	3,95	6,90	–
8085	4,95	7,00	2,0
Z80	2,45	4,90	1,8
Z80A	2,95	4.40	3,0
Z80B	8,95	7,10	5,0
6502	4,95	13,95	–
6502A	5,95	14,95	–
8086	24,95	23,00	8,0
8088	19,95	19,50	12,0
8089	59,90	–	–
8087-2	119,00	529,00	–
68000-8	19,95	49,50	28,0
68008-8	–	53,90	–
68010L8	–	169,00	–
	Pamięci EPROM		
2716	3,95	8,50	2,5
2732	4,25	8,20	4,0
2764	4,95	5,90	5,5
27128	7,50	6,80	7,0
	Pamięci RAM		
2102	0,89	4,25	–
2114	1,24	5,95	1,5
6116	3,69	3,50	3,5
6264	10,25	7.85	9,0
4116	0,87	3,25	0,8
4164	1,66	2,95	1,5
41256	5,95	9,95	–
	Sterowniki dysków elastycznych		
8271	69,95	–	–
8272	19,95	–	10,0
uPD765	19,95	16,90	8,0
179x	21,50	25,95	–
279x	32,95	27,30	25,0
	Sterowniki CRT		
8275	28,50	–	12,0
6845	11,95	11,00	–
uPD7220	34,95	39.90	–
EF9366	–	66,00	–
	Inne układy		
Z80A PIO	3,95	4,50	2,0
Z80A CIC	3,95	4,50	2,0
Z80A SIO	10,95	13,90	–
8212	1,80	6,50	0,8
8216	1,75	4,50	0,3
8251A	4,49	5,90	2,5
8253	6,95	5,90	–
8255	4,49	4,90	1,5
8282	6.50	7.50	–
8286	6,50	7,70	–

Sprzęt komputerowy	RFN	GB	POLSKA przedsiębiorstwa		POLSKA rynek
Atari 800XL	248	129	150	(P)	–
Atari 800XL+dysk	800	229	–	(P)	–
Amstrad CPC 464 (C)	1200	260	750	(B)	350
Amstrad CPC 6128 (C)	–	350	1150	(B)	500
Commodore C64	498	130	280		140
ZX Spectrum	298	70	150		80

JAK PISAĆ DO MIKROKLANU...

MIKROKLAN jest dla Ciebie. Oznacza to, że i od Ciebie zależy co będzie w nim drukowane. Będziemy wdzięczni, jeżeli napiszesz dlaczego Ci się to czy tamto nie podoba, a szczególnie wdzięczni, gdy zaproponujesz co robić, żeby było lepiej. Pamiętaj jednak, że papier nie jest z gumy i mieści się na nim skończona ilość informacji. Weź też pod uwagę prawdopodobną opinię pozostałych 99 999 Czytelników (niektórzy spośród nich są bez mała ekspertami, inni zaś stawiają pierwsze kroki...). Nadsyłane nam propozycje tematów, które zdaniem Czytelników powinny znaleźć się na łamach MIKROKLANU, kwalifikować będziemy według liczby zainteresowanych. Może właśnie Twój list przeważy szalę? Będziemy starali się reagować jak najszybciej, jednak droga od pomysłu do kiosku RUCH-u zajmuje kilka miesięcy.

Jeśli uważasz, że wiesz coś ciekawego i chciałbyś podzielić się tym z pozostałymi Czytelnikami – napisz artykuł. Jak wskazuje nazwa naszego pisma pragniemy tworzyć wspólnie z Czytelnikami klan ludzi, którym zależy, aby mikrokomputery były wykorzystywane z pełną świadomością ich możliwości i ograniczeń. Możesz więc napisać o tym, co sam odkryłeś lub udoskonaliłeś, posługując się Spectrum, Commodorem, Amstradem czy IBM PC (jeśli zainspirował Cię pomysł już gdzieś publikowany, powołaj się na niego, załączając do tekstu wykaz literatury). Możesz napisać o możliwościach rozbudowy lub bardziej efektywnym sposobie programowania. Możemy też opublikować Twój program, jeśli będzie on interesujący dla innych i... niezbyt długi. To samo dotyczy konstrukcji sprzętowych. Ciekawe rozwiązania układowe, niestandardowe aplikacje, czyli forum do wymiany pomysłów i doświadczeń dla wszystkich, hobbistów i profesjonalistów.

Wymagania dotyczące przysyłanych materiałów:

tekst: w maszynopisie, z dwoma kopiami, (30 wierszy na stronie, 60 uderzeń w wierszu, czyli duży odstęp między liniami)

objętość: maksymalnie 7-9 str. maszynopisu (jasno i zwięźle)

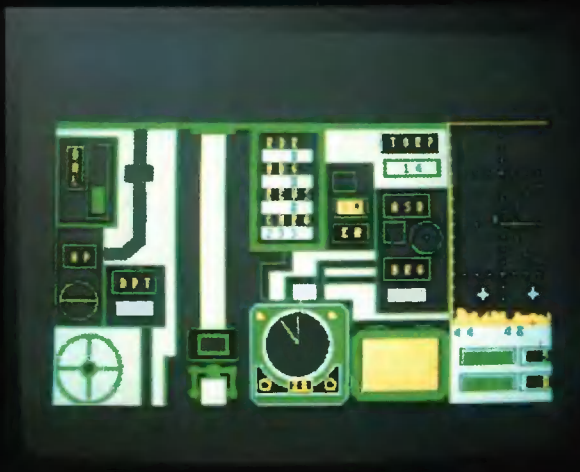
rysunki: mogą być w ołówku (przejrzyste i czytelne)

wydruki: maksymalnie kontrastowe (załóż nową taśmę do drukarki, jeśli możesz przyślij program na nośniku magnetycznym – zwrócimy!)

programy: nie tylko przetestowane, ale też eleganckie

konstrukcje: wyłącznie uruchomione i koniecznie sprawdzone

ponadto: adres domowy, telefony – domowy i służbowy, ew. numer konta PKO.



AMSTRAD STORY

czytaj na stronie 14-17

W NASTĘPNYM ZESZYCIE

ATARI